

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Программалық инженерия кафедрасы

Сейдазимов Сырым Байболұлы

Ұялы телефондар мен компьютерлерге арналған маркеттің клиенттік бөлігін
құру

Дипломдық жобаға
ТҮСІНІКТЕМЕЛІК ЖАЗБА

5B060200 – «Ақпараттану» мамандығы

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

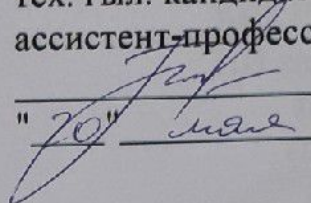
Программалық инженерия кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ

ПИ кафедра меңгерушісі

тех. ғыл. кандидаты,

ассистент-профессор

 Р. Юнусов

" 20 " _____ 2019ж.

Дипломдық жобаға

ТҮСІНІКТЕМЕЛІК ЖАЗБА

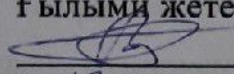
Тақырыбы: Ұялы телефондар мен компьютерлерге арналған маркеттің
клиенттік бөлігін құру

5B060200 – «Ақпараттану» мамандығы

Орындаған

Сейдазимов С.Б.

Ғылыми жетекші лектор

 Д. А. Баймбетов

" 17 " _____ 05 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

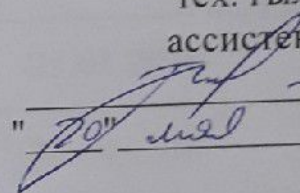
Ақпараттық және телекоммуникациялық технологиялар институты

Программалық инженерия кафедрасы

5B060200 – «Ақпараттану»

БЕКІТЕМІН

ПИ кафедра меңгерушісі,
тех. ғыл. кандидаты,
ассистент-профессор

 Р. Юнусов
"20" _____ 2019ж.

**Дипломдық жоба орындауға
ТАПСЫРМА**

Білім алушыға Сейдазимов Сырым Байболұлы

Тақырыбы: «Ұялы телефондар мен компьютерлерге арналған маркетинг клиенттік бөлігін құру».

Университет ректоры бұйрығының № 1828-б "13" наурыз 2019 ж. шешімімен бекітілген.

Орындалған жобаның өткізу мерзімі "21" мамыр 2019 ж.

Дипломдық жобаның бастапқы мәліметтері: Ұсынылатын дипломдық жобада ұялы телефондар мен компьютерлерге арналған маркетинг клиенттік бөлігін құру. Құру барысында JavaScript бағдарламау тілінің React библиотекасын қолдандым.

Есеп – түсініктеме жазбаның талқылауға берілген сұрақтардың тізімі:

- а) қазіргі заманғы веб-технологияларды зерттеу;
- б) дипломдық жобаға ұқсас веб-сайттардың дизайнын зерттеу;
- в) жобаны модельдеу, UML диаграмма құру;
- г) веб-сайтты құру, кітапханалармен жұмыс істеу, компоненттерді қосу.

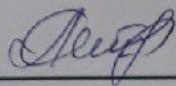
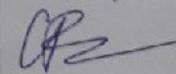
Графикалық материалдар тізімі (міндетті суреттердің нақты көрсетілуімен): презентацияның 20 слайдпен берілген құжат түрінде ұсынылған.

Ұсынылған негізгі әдебиеттер: 12 пайдаланылған әдебиеттер тізімінен.

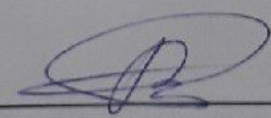
Дипломдық жобаны орындау
КЕСТЕСІ

Бөлімдердің атаулары, зерттелген мәселелердің тізімі	Ғылыми жетекшіге және кеңесшілерге ұсыну мерзімі	Ескерту
1. Дипломдық жобаның жоспарын құру.	15.01.2019	моу
2. Қазіргі заманғы веб-технологияларды зерттеу.	17.01.2019	моу
3. Дипломдық тақырып бойынша ғылыми теориялық материалдарды жинау.	02.02.2019	моу
4. Жобалау сызбаларын дайындау және веб-сайтты құру.	14.02.2019	моу
5. Жобаның веб-қосымшасын тестілеуден өткізу.	25.03.2019	моу
6. Дипломдық жобаға түсіндірме жазба жазуды аяқтау.	26.04.2019	моу

Дипломдық жұмыс бөлімдерінің кеңесшілерінің аяқталған жұмысқа қойған
қолтаңбалары

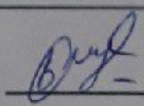
Бөлімдер атауы	Кеңес берушілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қолтаңба қойылған мерзімі	Қолы
Нормалық бақылаушы	Алғожаева Р. Сениор-лектор	16.05.2019	
Бағдарламалық бөлім	Сман Н. О. Ассистент	16.05.2019	

Ғылыми жетекші



Д. А. Баймбетов

Тапсырманы орындауға қабылдап алған студент



С. Б. Сейдазимов

Күні

«17» 05 2019ж.

Отчет подобия



Университет:	Saitbayev University
Название:	Сейдахимов Сырым ИнсБЕ-15 ТК 0001
Автор:	Сейдахимов Сырым
Координатор:	Дәулет Баймбетов
Дата отчета:	2019-05-05 12:34:25
Коэффициент подобия № 1: ☐	14,9%
Коэффициент подобия № 2: ☐	12,6%
Длина фразы для коэффициента подобия № 2: ☐	25
Количество слов:	6 352
Число знаков:	51 811
Адреса пропущенные при проверке:	

ҒЫЛЫМИ ЖЕТЕКШІ

Д.А. Баймбетов

«Программалық инженерия»

кафедрасының лекторы «17» 05 2019 жыл

АҢДАТПА

Дипломдық жобада заманауи технологияларды қолдана отырып ұялы телефондар мен компьютерлерді сатуға және сатып алуға арналған «KazMarket» жобасының веб-сайтын құру қарастырылған.

«KazMarket» веб-сайты қолданушыларға тіркелуге, ұялы телефондар немесе компьютерлер хабарландыруларын оқуға, хабарландыру астына пікір қалдыруға, сондай-ақ қолданушылар арасында хабарлама алмасуға мүмкіндік береді.

«KazMarket» веб-сайтын құру барысында Atom бағдарлама құру ортасы, JavaScript бағдарламалау тілі және React кітапханасы қолданылды.

Жобаны құру барысында пайдаланушы интерфейсін барынша түсінікті және қарапайым жасауға ден қойылды. Осыған орай еліміз аясындағы басқа тақырыпқа негізделген, бірақ өз кезегінде хабарладыру тақтасы болып табылатын Olx, Market сынды қосымшалармен салыстырылып артықшылықтары мен кемшіліктері ескерілді. Бұл салыстырулар пайдаланушыларға үйреншікті элементтерді қамтитын қосымша құру мақсатында жүргізілді.

АННОТАЦИЯ

Дипломный проект предусматривает создание веб-сайта для проекта «KazMarket» для поиска мобильных телефонов и компьютеров с использованием новейших технологий.

Веб-сайт «KazMarket» позволяет пользователям регистрироваться, читать уведомления об утрате или обнаружении, публиковать комментарии, оставлять отзывы. При создании веб-сайта «KazMarket» использовались среда создания приложений Atom и язык программирования JavaScript а также библиотека React.

При проектировании пользовательский интерфейс был разработан так, чтобы быть максимально простым и легким. В этой связи были учтены преимущества и недостатки других вопросов внутри страны по сравнению с преимуществами Olx, Market, которые являются карточными карточками. Эти сравнения были сделаны для создания дополнительных пользователей, которые включают знакомые элементы.

ANNOTATION

The graduation project provides for the creation of a web-site for the «KazMarket» project to search for mobile phones and computers using the latest technologies.

The «KazMarket» web-site allows users to register, read notifications of loss or detection, publish comments, and evaluate user and application messages.

When creating the «KazMarket» web-site, the Atom application creation environment and the JavaScript programming language and also React library were used.

When designing, the user interface was designed to be as simple and easy as possible. In this regard, the advantages and disadvantages of other issues within the country were taken into account in comparison with the advantages of Olx, Market, which are card cards. These comparisons were made to create additional users, which include familiar elements.

МАЗМҰНЫ

	Кіріспе	8
1	Жалпы бөлім	9
1.1	Веб-сайт туралы түсінік	9
1.2	Ұялы телефондар мен компьютерлерге арналған маркеттің мақсаты, міндеті	10
1.3	Ұялы телефондар мен компьютерлерге арналған маркетті құру үшін қолданылатын технологиялар	13
1.4	JavaScript тілінің кітапханалары мен фреймворктары	18
1.4.1	React	18
1.4.2	Vue	19
1.4.3	Angular	19
1.5	Веб-сайт құру барысында пайдаланатын құралдар мен жабдықтар	20
2	Жобалау бөлімі	23
2.1	UI әзірлеу әдістері мен құралдары	23
2.2	UML бірыңғай модельдеу тілі	24
2.3	Rational Rose құралы	25
2.4	Прецеденттер диаграммасы	26
2.5	Тізбек диаграммасы	28
2.6	Класстар диаграммасы	29
3	Қолданбалы бөлім	32
3.1	Ұялы телефондар мен компьютерлерге арналған маркеттің веб-сайты	32
	Қорытынды	36
	Пайдаланылған әдебиеттер тізімі	37
	А қосымшасы. Техникалық тапсырма	38
	Б қосымшасы. Бағдарлама мәтіні	44

КІРІСПЕ

Қазіргі таңда әлемде ақпараттық технологиялардың дамуына байланысты, адамдарға қолайлы болу үшін барлық жағдай жасалу үстінде. Мысалы, осыдан бірнеше жылдар бұрын белгілі бір зат сатып алу керек болса, міндетті түрде базар аралап кететін болған. Ал қазір заманның, яғни технологиялардың дамуына байланысты уйде, жұмыс орнында және кез-келген жерде отырып ақ кез-келген затты сатып алуға немесе тапсырыс беруге болады. Осындай мүмкіншіліктерді арттыру мақсатында ғаламторда көптеген сервистер мен web-сайттар көбею үстінде. Сонымен қатар, ол web-сайттардың мобильді қосымшасында IOS және Android платформаларында жасап шығаруда. Бұл да қолданушылардың сервисті қолдануы барынша ыңғайлы болуы үшін жасалады. Сондай бір сервистерді атап өтетін болсақ олар: OLX, Krisha, Kolesa, Market және т.б. Осы атап өткен сервис-платформалардың web-сайттарыда және мобильді қосымшалары да бар. Соның арқасында бұл сервистердің қолданушылары да жеткілікті. Олардың әрқайсысына жеке және қысқаша тоқталып кететін болсақ:

OLX (ағылш. Online Exchange – онлайн алмасу) – адамдардың сату/сатып алу үшін біріктіретін онлайн хабарландырулардың платформасы болып табылады.

Krisha, Kolesa, Market – жылжымайтын мүлік, көлік және басқа да көптеген өнімдер мен қызметтер үшін арналған тегін хабарландыру web-сайты.

Жоғарыда атап өтілген сервистер әртүрлі мақсаттарға қолданылады. OLX – ол барлық категориядағы тауарлар мен қызметтер платформасы болып табылады. Ал, Krisha – жылжымайтын мүлікке қатысты қызметтер, Kolesa – көлікке қатысты қызметтер, Market – барлық категорияда. Қазіргі таңда ең актуалды мәселелердің бірі ол ұялы телефондар және компьютерлер. Себебі әлем бойынша барлық дамыған мемлекеттер толығымен цифрлық технологияға ауысу барысында. Сондықтан да, қазіргі заманда адамға ең қажетті заттардың бірі ол – ұялы телефондар (смартфондар) мен компьютерлер болып табылады. Смарт сағаттарды да жатқызсақ қателеспейміз. Ал енді осы айтылған заттарға қызмет көрсететін сервис әлі жоқ десек те болады. Мысалы, белгілі бір адам ұялы телефон іздегісі келсе, басқа сервистерден де табуға болады әрине, дегенмен ол адамға қолайсыз. Себебі, басқа тауарлар адамның іздеген затына кедергі келтіреді. Менің де жасап жатқан жұмысымның мақсаты – осы мәселелерді шешу болып табылады.

1 Жалпы бөлім

1.1 Веб-сайт туралы түсінік

Осы заманда ақпараттық технологиялардың өркендеуіне байланысты мыңдаған сервистер енгізілуде.

Цифрлық технологияға дейінгі уақытта ақпарат алмасу немесе хабарлау қазіргі жағдайдан әлдеқайда қолайсыз болды. Ақпараттық технологиялардың дамуына байланысты бүгінгі таңда ақпарат тарату немесе хабарландыру жарықтың жылдамдығымен бірдей десек те болады. Себебі қазір ғаламтордың арқасында желіде web-сайттар мен сервистердің саны жетерлік.

Кез-келген жерде, көшеде немесе жұмыс орнында, теледидар және радиодан сайт деген сөзді өте жиі естіп жатамыз. Бірақ шын мәнісінде оның не екені біреуге мәлім біреуге жоқ. Сондықтан да сол сөздің түпкірінде не жатқанын түсіндіріп, тоқталып кетуді жөн көріп отырмын. Сайт – қарапайым тілмен айтқанда ғаламдық паутинада орналасқан жай ғана парақша. Дегенмен, сол парақша бізге қалай көрінеді немесе қалай беріледі ол қарапайым қолданушыларға әрине беймәлім. Сайтқа кіру үшін пайдаланушылар әдетте веб-браузер деген атауға ие арнайы программалар қолданады. Олар: Internet Explorer, Google Chrome, Mozilla Firefox, Opera, Safari және т.б. Белгілі бір бетті көру үшін сол программа ішіндегі жолға адресі енгіземіз және браузер бізге сұрап тұрған бетті көрсетеді. Браузер біз адресі енгізген бетте ол адресі Интернеттің көмегімен басқа біз сұрап жатқан файлдар орналасқан арнайы компьютерге (веб - сервер) сұрау жібереді. Веб-сервер – интернетке тұрақты түрде қосылуы тұрған, сұрауларға арнайы ыңғайланған компьютерлер. Кейбір үлкен компаниялар өздерінің жеке веб-серверін пайдаланады. Ал қалғандары белгілі бір ақшаға хостинг компанияларға жүгінеді. Біздің жасаған сайттарымыз сол компаниялардың серверінде орналасады. Және ол серверлердің жылдамдағы, жадысы өте маңызды. Себебі сайтқа қолданушылар телефоннан, компьютерден, планшеттен кіреді. Ал олардың әрқайсысында жылдамдықтары әртүрлі. Айта кететін жайт, сайт орналасқан сервер әлемнің кез-келген жерінде болуы мүмкін. Оның орналасқан жерін анықтау үшін браузер міндетті түрде DNS-серверге (Domain name system) жүгінеді.

Интернетке қосылу провайдер арқылы жүзеге асырылады. Белгілі бір сайтқа кіру үшін сілтеме жолына сайттың атын енгізу керек. Мысалы, www.google.kz. Содан соң компьютер DNS-серверден тұратын желіге қосылып, IP-адресі анықтайды. IP-адрес 12 саннан аспайтын, нүкте арқылы ажыратылған 4 бөліктен тұрады. Кез-келген желіге қосылған құрылғының өзінің жеке IP-адресі болады. Жалпы веб-сайт ғаламтор паутинасында осылай жұмыс істейді.

1.2 Ұялы телефондар мен компьютерлерге арналған маркеттің мақсаты, міндеті

Ұялы телефондар мен компьютерлер қазіргі заманда ең көп қолданылатын, ең қажетті және ең актуалды десек болады. Технологиялардың дамуына байланысты әлемдегі көп сервистерді пайдалану үшін смартфон немесе компьютер қажет болады. Қазіргі таңда тіпті көптеген гаджеттерде бар.

Баяғы заманда қызметтің түрлері шектеулі болатын. Қазір қызметтің түрі өте көп. Және кез-келген қызметтің көбісі интернет арқылы да қызмет көрсетеді. Олар: интернет-магазиндер, хабарландыру сайттары, авиа қызметтер, такси қызметтері және т.б. Бұл сервистердің әрқайсысының өзіне сай веб-сайт немесе қосымшалары бар. Себебі ол қазіргі таңда актуалды әрі тиімді. Салыстыратын болсақ, баяғыда технологиялар қазіргідей дамымай тұрып адамдардың өмірі мүлде басқаша болатын. Мысалы, белгілі бір зат сатып алу керек болса, міндетті түрде базар немесе дүкендерге барып сауда жасайтын болған. Олай сауда жасаудың әрине пайдасы мен зияны бар. Пайдасы ол алғын келген тауарды ұстап көре аласын. Ал зияны сол тауарды көру үшін уақытынды жолға кетіресін. Авиабилет алу үшін баяғыда әуежайға немесе аэробекеттерге барған болса, қазір веб-сайт немесе программаның көмегімен өзіне лайықты күнге, қолжетімді бағаға онлайн түрде билет алуға болады. Осының бәрі технологиялардың дамуының арқасында десек қателеспейміз. Тіпті такси шақыру керек болса да, қосымшаның көмегімен ыңғайлы әрі тез такси шақыруға болады. Менің дипломдық жұмысымда осы айтылған мысалдар секілді, адамдарға барынша ыңғайлы қызмет көрсету.

Тағы да бір мысал қарастыратын болсақ, көлік таңдау немесе сатып алу. Әдетте көлікті көлікбазардан қарап, таңдап, сатып алатын болған. Бірақ заманауи технологиялардың дамуына байланысты көптеген программалар мен веб-сайттар бар. Сондай веб-сайттардың түрлерін айтатын болсақ, олар: kolesa.kz, avto.kz және т.б. Kolesa.kz Қазақстанда қолданылатын веб-сайт, ал avto.kz Ресейде. Kolesa.kz Қазақстанның сервис болғандықтан осы веб-сайт туралы көбірек ақпарат айтайық. Бұл сайт 1996 жылдан бері 2014 жылға дейін шығарылып келген “Колеса” газетінің негізінде құрылған. Сайт 2006 жылы құрылып, қазіргі таңда үлкен, ауқымды жобаға айналды. Бұл тек онлайн тегін хабарландырулар ғана емес, автокөлікшілерге арналған нағыз сауда орталығы және электронды мағлұтнама. “Колеса” жаңартуымен және қолдауымен кәсіби журналисттер, редакторлар, программистер айналысады. Сайттың қысқаша нұсқауын айтатын болсақ, сайттың мазмұнын екі бөлікке бөліп қарастыруға болады. Олар:

- жеке тұлғалар мен компаниялардың жарнамалары;
- автомобильдік материалдар.

Осы веб-сайттар пайда болғалы адамдардың өмірі жеңілдемесе, қиындай түспегені айдан анық. Бұл сайттар тек көлік саласына қатысты қызметтер

көрсетеді. Басқа салаларға да арналған түрлі-түрлі веб-сайттар жетерлік. Соның бірі ол – krisha.kz.

Сайттың атауынан ақ қай салаға арналғанын байқауға болады. Ол әрине үй, пәтерлерге арналған веб-сайт. Бұл сайт дәл kolesa.kz сайтына өте ұқсас, айырмашылығы тек саласында. Сайт үй және пәтер сатып алам деушілерге немесе сатушыларға өте қолайлы. Себебі сайтта пәтер іздеуге барлық жағдайлар жасалған. Мысалы, бөлме бойынша іздеу, бағасы бойынша және т.б. Сонымен қатар, сату мен сатып алудан басқа құрылыс және жөндеу, жобалау және жобалау, тасымалдау, тазалау, жиһаз және сантехникалық бұйымдарды монтаждау қызметтерінде көрсетіледі.

Сонда басқа заттарды қайдан сатып аламыз деген сұрақ қолданушыларда туындауы мүмкін. Бірақ ондай сұрақтарға да жауап беретін веб-сайттар бар. Олар: olx.kz, market.kz. Бұл сайттарға қысқаша анықтама беретін болсақ, бұл сайттар барлық салада, яғни категориядағы хабарландырулар. Себебі бұл сайттар кез-келген заттарды сатуға немесе сатып алуға болады. Дегенмен қазіргі уақытта көлік немесе пәтер сияқты ерекшелік керек заттарда бар. Олар әрине электроникаға қатысты тауарлар. Анығырақ айтсақ, ұялы телефондар және компьютерлер. Себебі ұялы телефондар мен компьютерлер пәтер мен көлік сияқты актуалды десек қателеспейміз. Статистика бойынша кез-келген екінші адамда ұялы телефон немесе компьютер бар. Өйткені заманның талабы солай. Тіпті жоғары айтып өткен kolesa.kz, krisha.kz секілді сайттардың өзін пайдалану үшін бізге ғаламторға кіру мүмкіндігі бар ұялы телефон немесе компьютер қажет болады. Бұдан бөлек адамдар өзара байланысу үшін, бір-бірімен хабар алмасу немесе жаңалықтармен бөлісу үшін де ұялы телефондарды сатып алады. Бүгінгі таңда ұялы телефондар мен компьютерлердің түрі өте көп. Сол үшін де бұл заттарды арнайы сайтқа жасау ыңғайлы болып табылады. Мысалы, белгілі бір адам ұялы телефон алғысы келсе дүкен аралап жүрмей, онлайн түрде телефондарды бір-бірімен салыстырып өзіне ұнағанын алады. Olx.kz, market.kz деген сайттардан ақ телефон немесе компьютерді алуға болады ғой, неге соны пайдаланбасқа деген сұрақ міндетті түрде туындайды. Бірақ ол сайттар телефондар мен компьютерлерге арналған тематика болмағандықтан адамдарға өзінің іздеген телефонын табу қолайсыздық тудырады. Себебі ол сайттарда тек телефондар мен компьютерлер емес барлық заттың түрлері бар. Сондықтан да қолданушы іздеген затты ұмытып қалмауы ғажап емес.

Менің жасап жатқан сайтым осы артықшылықтарды арттырып, кемшіліктерді азайту мақсатында болып табылады. Себебі менің жасайтын сайтымда, тек қана ұялы телефондар мен компьютерлер және оларға арналған қызметтер болады. Байқасаңыздар жоғарыда мысал ретінде айтып өткен сайттарда сол салаға қатысты қызметтері бар. Мысалы, kolesa.kz сайтында көлікке қатысты қызметтер ұсынылады. Олар: көлік жөндеу, жалға беру немесе алу және т.б. Ал krisha.kz сайтында: құрылыс және жөндеу, жобалау және жобалау, тасымалдау, тазалау, жиһаз және сантехникалық қондырғылар. Сол сияқты міндетті түрде сайтта қызметтер жүйесі кіреді. Ол жерде ұялы телефон немесе компьютерлерді жөндеу сияқты қызметтер көрсетіледі. Себебі

дәл осы қызмет біздің сайтты ерекшелей түседі. Қарапайым мысал қарастыратын болсақ, белгілі бір адамның ұялы телефоны істен шығып қалған жағдайда ол адам қайда баратынын білмей өзінің жүйкесін тоздырып және өзінің алтын уақытын кетіреді. Ал біздің сайттың көмегімен ол адам оңай қызмет көрсететін жерді тауып алады. Бұл жасап жатқан сайттын тек қана қолданушыларға емес, қызмет көрсету орталықтарына да пайдасы өте зор. Себебі осы сайт арқылы олар өздерінің қызмет орталығын жарнамалай алады. Сондай-ақ қолданушылар қызмет орталығын бағалай алады. Ол әрине тағыда қолданушыларғы ыңғайлылықты арттыра түседі. Өйткені, қызмет орталығының қаншалықты жұмысын жақсы істейтінін сол бағаға қарап анықтауға болады. Сонымен қатар сайтта категория бойынша іздеу, бағасы бойынша іздеу, модель бойынша іздеу, сұрыптау және т.б жағдайлар қарастырылған. Себебі, ұялы телефон немесе компьютер іздеген адам категория бойынша, модель немесе баға бойынша іздейді. Осы айтылған нәрселердің бәрін тек артықшылық ретінде қарастыруға болады. Кемшілікке тек бір ғана нәрсені жатқызуға болады, ол – интернетке кіру мүмкіншілігінің қажет болуы. Бұның өзін кемшілік ретінде қарастыру өте қиын. Себебі қазіргі заманда интернеттің болмауы өте сирек кездеседі және кейбір аудандарға ғана. Ал басқа жердің барлығында барынша қолжетімді. Енді қолданушылар өзінің заттарын қалай хабарландыруға береді? Хабарландыру беру үшін кез-келген қолданушы сайтқа міндетті түрде тіркеліп, өзі туралы қысқаша ақпарат енгізуі керек. Бұл әрекет барлық қолданушылардың қауіпсіздігі үшін және әр қолданушы өзінің жеке кабинетіне кіру үшін және өзінің хабарландыруларын көру үшін жасалады. Ары қарай қолданушы өзінің тауарының тақырыбын көрсетіп, тауары жайлы қысқаша сипаттама беріп, категориясын таңдап, фотосуреттерді қосып және бағасын көрсету арқылы хабарландыру береді. Фотосуреті бар тауарлар өзіне көп назар аударатындығын және сатылу ықтималдылығының жоғары екендігін айта кету керек. Себебі сатып алушы адам ең алдымен тауардың түрін көргісі келеді. Сондықтан да хабарландыру беру сайттарында әрдайым хабарландыруды суретпен берген жөн. Айта кететін жайт, жоғарыда айтып өткен сайттар (kolesa.kz, krisha.kz, olx.kz, market.kz) және менің жасап жатқан веб-сайтымда хабарландыру беру тегін болып табылады. Алайда, егер хабарландыру беруші адам немесе жеке компания өзінің тауарының тезірек сатылып кеткенін қаласа, сайтта хабарландыруды ерекшелеу немесе жоғарыға көтеру секілді қызметтері бар және ақылы түрде. Дегенмен бұл қызметтер қолданушыны осы қызметтермен қолдануды міндеттемейді. Баяғы заманмен салыстыратын болсақ, бұл жерде сайттын артықшылығын және ыңғайлылығын тағыда байқауға болады. Себебі баяғыда хабарландыру беру газет арқылы жүзеге асатын болған және ол қызмет ақылы түрде болатын. Және ақылы болсада басқа амалы болмағандықтан сол үшін адамдар уақыттарын құртып, хабарландыру беру орталықтарына барған.

1.3 Ұялы телефондар мен компьютерлерге арналған маркетті құру үшін қолданылатын технологиялар

Веб-сайт саласы өте кең болғандықтан, сайт жасау барысында қолданатын құралғыларды таңдауға таңдау өте көп. Бірақ кез-келген веб-сайттың негізі ол – html, css, javascript болып табылады. Ал егерде сайт барынша функционалды әрі ыңғайлы және адаптивті болу үшін міндетті түрде қосымша кітапханалармен фреймворктарды пайдалануға тура келеді. Жалпы сайттың негізі деп өзім айтып өткен нәрселерге қысқаша анықтама беріп кетейін.

Веб-сайт – (ағылшынша Website: Web – тор, желі және site – орын, желідегі орын, сегмент) – бір мекенжайға (домен атымен немесе IP – мекенжаймен) біріктірілген компьютер желісіндегі жеке тұлғаның немесе ұйымдары электрондық құжаттар (файлдар) жиынтығы. Ғаламтор желісінің негізін құрайтын барлық сайттар болып табылады. Сервердегі сайтқа қолданушының бірден қол жеткізуі үшін арнайы HTTP хаттамасы жасалған.

HTTP хаттамасы – Web-сервер мен клиент арасында мәліметтерді жіберудің негізгі, әрі аса қарапайым тәсілі. Web және HTTP пайда болмай тұрып, Интернетте файлдарды жіберу үшін енгізу/шығару (input/output, I/O) хаттамасы ретінде көбінесе FTP қолданылды. HTTP – бұл URL-мен жұмыс істейтін және гипермәтін/гипермедиа орталарына арналған тұтас, әрі шапшаң енгізу/шығару хаттамасы. FTP-ге қарағанда, бұл күйсіз хаттама және оның бірнеше бұйрығы (тәсілі) ғана бар. MIME-ді қолданғанының арқасында, HTTP мәліметтердің көптеген пішімдеріне және түрлі енгізу/шығару міндеттеріне бейімделеді. HTTP – сұрау/жауап үлгісін іске асыратын клиент-серверлік хаттама. HTTP-клиент, немесе пайдаланушылық агент (әдетте бұл Web-браузер), HTTP-серверге URL көмегімен қосылады да, қорды сұратады, мысалы HTML-құжатты. Мәліметтердің инкапсуляциясы үшін бұл үлгіде MIME кеңейтілімдері қолданылады. Клиент пен сервер арасында жіберілетін мәліметтер құрылымы электрондық поштаны еске түсіреді. Ол хабарлама денесінен және метамәліметтерден (хабарламалардың тақырыптарынан) тұрады. HTTP хаттамасы MIME пішіміндегі ақпаратты жібереді. Метамәліметтерге HTTP-сервер мен клиент арасында мәліметтерді жіберуге қажетті ақпарат кіреді. Дегенмен, HTTP әдеттегі MIME рұқсат етпейтін (пошталық шлюздердің 7-биттік шектеулері үшін) екілік пішімді ұйғарады. Байланыс сеансына, әдетте, HTTP-клиенттер (Web-браузерлер) және HTTP-серверлер (Web-серверлер), және кейде – прокси-серверлер қатысады. Соңғылары клиенттің негізгі сұрауын шлюз (мысалы, компанияның интражелісі мен Интернет арасындағы желіаралық экран) арқылы жібере отырып, клиентке қатысты сервер және басқа серверге қатысты клиент ретінде шығады. Әдетте, HTTP-клиенттер мен серверлер HTTP үшін дағды бойынша сақталған TCP/IP-дің 80-ші порты арқылы қатынасады. Дегенмен, URL-де айқын нұсқалған басқа да порттар қолданылуы да мүмкін. Толықтыра келе, HTTP нақты TCP/IP-ді қолдануды алдын ала ойламайды және кепілді жеткізіп берудің басқа да хаттамаларымен де жұмыс істейді байқаймыз.

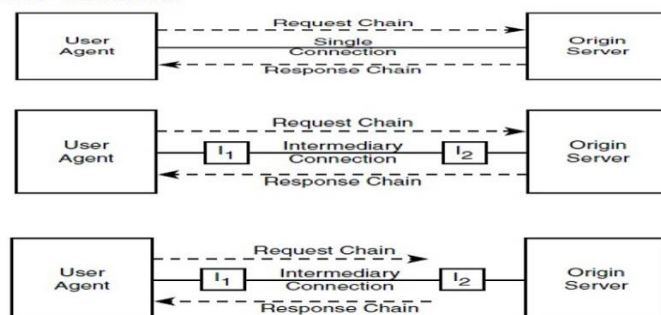
Web-Браузер жиі көптеген объектінен құралған, мысалы HTML-құжаттың өзін және бірнеше сурет (GIF, JPEG, PNG және т.б.) Web-парақтарды қарап шығады. HTTP-клиенттердің көбісі негізгі HTTP-құжаттың оқылуы үшін тек бір ағынды туғызады (серверге тек бір қосылумен), содан кейін қалған қажетті файлдарды алу үшін тағы бірнеше ағындарды (әрбіреуі серверге жеке қосылумен) туғызады. Қосылу клиенттің сұрауы бойынша орнатылады және сервердің жауабынан кейін үзіледі.

HTTP хаттамасының хабарламалары, әдетте HTTP-клиенттердің сұрауы мен HTTP-серверлердің жауабы болып табылады. Сұрау-хаттамаларда сұраудың өзі кіретін «сұрау жолы» (Request-Line), ал жауап-хаттамаларда жауаптың өзі кіретін, және де негізгі мәліметтер кіретін хаттаманың денесі (немесе мәні) кіретін «қалып күй жолы» (Status-Line) бар. Еңгізу/шығарудың басқа хаттамаларымен салыстырғанда HTTP-ге аз бұйрықтар (немесе тәсілдер) кіреді. Міндетті жүзеге асыруға тек үш тәсіл жатады – GET, HEAD және POST. Қалған төртеуі – PUT, DELETE, LINK және UNLINK – сипатталған, бірақ олар сонша түсінікті емес. Бұл тәсілдерге қысқаша түсінік беріп кететін болсам, олар:

- GET – құжат сұранысы. Анағұрлым көп қолданылатын әдіс;
- HEAD – құжат тақырыбына сұраныс. GET-тен, тек құжат жөнінде ақпарат жазылған тақырып сұранысы шығарылатындығымен ерекшеленеді. Құжаттың өзі шығарылмайды;
- POST – бұл әдіс мәліметтерді CGI-скрипттеріне жіберу үшін қолданылады. Мәліметтердің өзі келесі қатарларда параметрлер түрінде орналасады;
- PUT – құжатты серверге орналастыру. Сирек қолданылады. Бұл әдіс қолданылған сұраныс құжаттың өзі жіберілетін бөліктен тұрады.

HyperText Transfer Protocol (HTTP)

HTTP Sessions



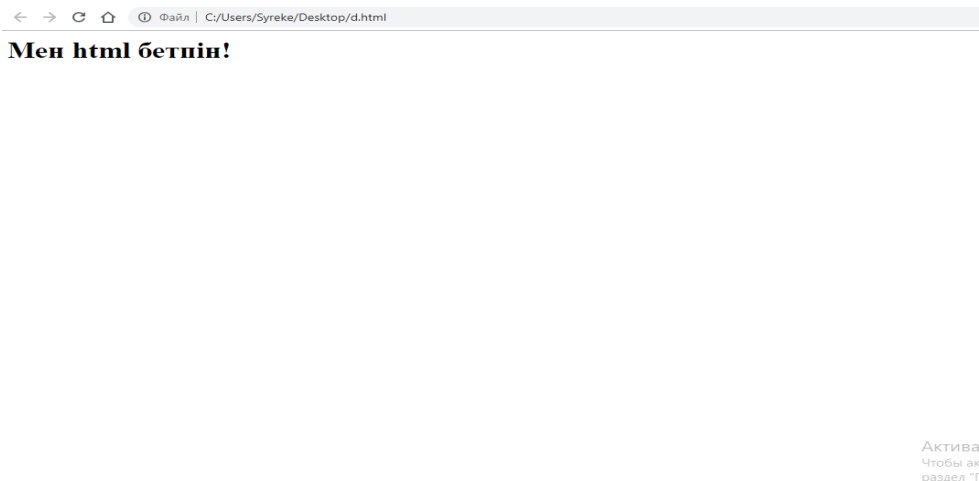
1.1-сурет – HTTP хаттамасының жұмыс жасау схемасы

Сайт – домен, хостинг және қозғалтқыштан (cms) тұрады. Кез-келген сайттың беті HTML (Hypertext markup language) беттерден тұрады.

HTML – ағылшын тілінің Hyper Text Markup Language деген сөздерінің аббревиатурасы, яғни html дегеніміз гипер текстті белгілеу тілі деп қазақшаға аударылады. HTML сайт жасау үшін қолданылатын тіл. Ғаламтордағы мәліметерді бейнелеу HTML тілінде жазылады. Бірақ программалауға ұқсас болғанмен программалау тілі емес, гипермәтінді белгілеу тілі деп аталады. HTML – компьютерлер арасындағы мәліметтер алмасудың жаңа хаттамасы. Ол Интернеттегі мәтін бөліктерінің атқаратын қызметін белгілеп, оларды әрбір тұтынушыға жеткізе алатын құжатты түрде белгілейтін тіл болып табылады. HTML элементтерін мына категорияларға топтауға болады:

- тақырыптық элементтер;
- құжат құрылымын сипаттайтын элементтер;
- мәтіндік элементтер.

HTML құжаты сол құжаттың негізгі мәтінінен және белгілеу тәгтерінен тұрады да, қарапайым символдар жиыны болып табылады.



← → ↻ 🏠 ⓘ Файл | C:/Users/Syreke/Desktop/d.html

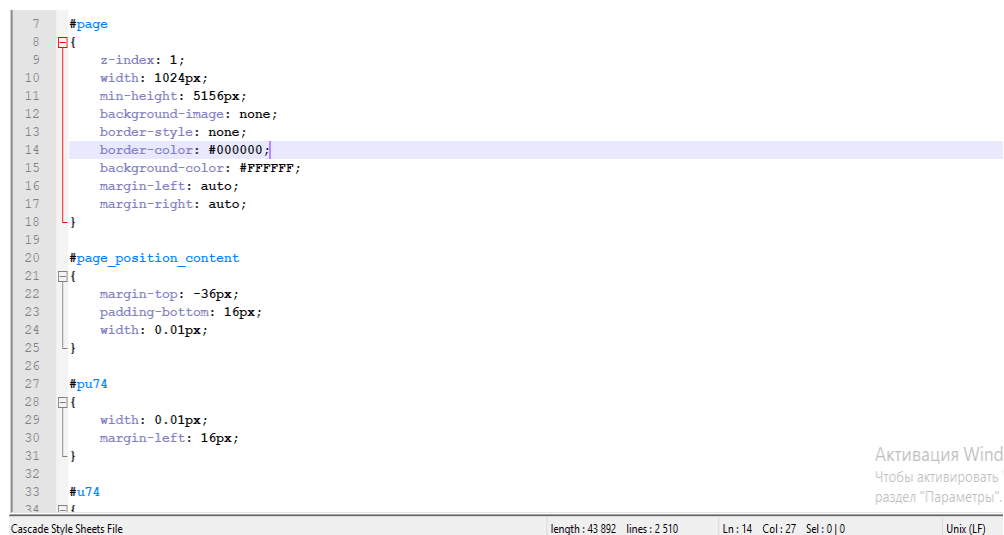
Мен html бетпін!

Актив
Чтобы ак
раздел "Г

1.2-сурет – HTML беттің көрінісі

Суретте байқағандай, жай ғана html бізге әдемі көрініс және дизайн бере алмайды. Сондықтан да міндетті түрде осындай жағдайда CSS (Cascading Style Sheets) көмекке келеді.

CSS – ағылшынша Cascading Style Sheets , яғни қазақша стилдік каскад таблицалары деп аталады. Қарапайым html – да сайт жасайтын болсақ, сайтымыз ыңғайсыз және көріксіз болып қалады. CSS арқылы біз веб беттерге стиль бере аламыз. Стильдерді селектор арқылы береміз. Селектордың 3 түрі бар. Олар: тәг арқылы, класс арқылы және id арқылы. Осы 3 тәсілді пайдалана отырып, веб сайтымызды әрлей аламыз.



1.3-сурет – CSS синтаксисі

Дегенмен, толыққанды веб-сайт құру үшін бұл екі құралда жеткіліксіз. Себебі html және css сайттың тек статикалық жағын береді. Ал веб-сайтқа функционалдылық беру үшін JavaScript қажет болады.

JavaScript – бұл салыстырмалы қарапайым объектілі – бағытталған тіл, және ол үлкен емес клиенттік және серверлік қосымшаларды Интернет үшін жасауға және құруға арналған. JavaScript тілінде жазылған бағдарламалар HTML-құжаттамалары құрмына қосылады және солармен бірге таралады. Көру бағдарламалары (ағылшын тілінің browser сөзінен шыққан браузерлер) Netscape Navigator және Microsoft Internet Explorer типтері бағдарлама-қоюшы құжаттамаларында тұрғызылған құжат мәтіндерін таниды (script-коды) және оларды орындайды. Олай болса, JavaScript – бағдарламалаудың интерпретацияланатын түрі. JavaScript жасалған бағдарламалардың мысалы ретінде пайдаланушымен енгізілген мәліметтерді тексеретін немесе құжаттаманы ашу немесе жабу кезінде қандай да бір әрекеттерді орындауға атқаратын бағдарламаларын қарастыруға болады. Мұндай бағдарламалар негізінен пайдаланушының әрекетіне әрекет ете алады – тышқан пернесін басу, экранды формаға мәліметтерді беру немесе тышқанды жылжыту арқылы беттерде жылжуға мүмкіндік береді. Одан басқа, JavaScript-бағдарламалар браузерлердің өзін және құжаттамалар атрибуттарын басқара алады.

JavaScript тілі синтаксисі бойынша Java тілімен ұқсас бола отырып, объектілі модельді қоспағанда, өз кезегінде мәліметтердің статикалық типтері және қатаң типтеу қасиеттеріне ие емес. JavaScript бағдарламасында Java бағдарламасынан өзгешелігі, кластар түсінігі тілдің синтаксистік конструкциясының негізі болып табылмайды. Мұндай негіз болып мәліметтердің алдын ала анықталған типтерінің қатары, яғни атқарылатын жүйемен көтерілетін типтері: сандық, булевік және қатарлық есептеледі, сонымен қатар өз бетінше дербес функцияларды орындаушылар және объектілер әдістері (JavaScript терминологиясындағы әдістер – және сол сияқты басқалар, яғни функция/бағдарламалар); өз қасиеттеріне және әдістеріне ие алдын ала

анықталған объектілер үлкен қатарына ие объектілі модель және т.б., сонымен қатар жаңа объектілерді пайдаланушылар үшін бағдарламалардағы тапсырмалар ережелері қатары және т.б. жатады.

JavaScript бағдарламаны жасау үшін ешқандай қосымша құралдар қажет емес, тек қана сәйкес версиядағы JavaScript тілін көтере алатын және HTML – құжаттамаларды құруға мүмкіндік беретін мәтіндік редактор болса болды. Бағдарлама JavaScript тікелей HTML-құжаттама мәтінінде тұрғызылатын болғандықтан сіз өзіңіздің жұмысыңыздың нәтижесін құжаттаманы браузермен көру кезінде көре аласыз және қажет болғанда өзгерістер енгізуге мүмкіндігіңіз болады. JavaScript тілі синтаксисі бойынша Java тілімен ұқсас бола отырып, объектілі модельді қоспағанда, өз кезегінде мәліметтердің статикалық типтері және қатаң типтеу қасиеттеріне ие емес. JavaScript бағдарламасында Java бағдарламасынан өзгешелігі, кластар түсінігі тілдің синтаксистік конструкциясының негізі болып табылмайды. Мұндай негіз болып мәліметтердің алдын ала анықталған типтерінің қатары, яғни атқарылатын жүйемен көтерілетін типтері: сандық, булевтік және қатарлық есептеледі, сонымен қатар өз бетінше дербес функцияларды орындаушылар және объектілер әдістері (JavaScript терминологиясындағы әдістер — және сол сияқты басқалар, яғни функция/бағдарламалар); өз қасиеттеріне және әдістеріне ие алдын ала анықталған объектілер үлкен қатарына ие объектілі модель және т.б., сонымен қатар жаңа объектілерді пайдаланушылар үшін бағдарламалардағы тапсырмалар ережелері қатары және т.б. жатады.

JavaScript бағдарламаны жасау үшін ешқандай қосымша құралдар қажет емес, тек қана сәйкес версиядағы JavaScript тілін көтере алатын және HTML – құжаттамаларды құруға мүмкіндік беретін мәтіндік редактор болса болды. Бағдарлама JavaScript тікелей HTML-құжаттама мәтінінде тұрғызылатын болғандықтан сіз өзіңіздің жұмысыңыздың нәтижесін құжаттаманы браузермен көру кезінде көре аласыз және қажет болғанда өзгерістер енгізуге мүмкіндігіңіз болады.

```
7 // to show the file browser dialog
8 $(this).parent().find('input').click();
9 });
10
11 // Initialize the jQuery File Upload plugin
12 $('#upload').fileupload({
13
14     // This element will accept file drag/drop uploading
15     dropZone: $('#drop'),
16
17     // This function is called when a file is added to the queue;
18     // either via the browse button, or via drag/drop:
19     add: function (e, data) {
20
21         var tpl = $('<li class="working"><input type="text" value="" data-width="48" data-height="48"
22             ' data-fgColor="#0788a5" data-readOnly="1" data-bgColor="#3e4043" /><p></p><span></span></li>');
23
24         // Append the file name and file size
25         tpl.find('p').text(data.files[0].name)
26             .append('<i>' + formatFileSize(data.files[0].size) + '</i>');
27
28         // Add the HTML to the UL element
29         data.context = tpl.appendTo(ul);
30
31         // Initialize the knob plugin
```

Ақ

1.4-сурет – JavaScript программалау тілінің синтаксисі

Бұл тілдің көмегімен көрінетін элементтермен және HTML – құжаттамалардың мазмұнымен динамикалы басқаруға болады. Экранда

қрнектелетін құжаттамаға шамаланып құрылған HTML құжаттамаларды браузермен жүктелген құжаттамаларды синтаксистік талдау процесінде жазуға болады. Document объектісінің көмегімен құжаттамаларды «нольден бастап» генерациялауға болады, ол негізінен пайдаланушының жасаған алдыңғы әрекеттеріне немесе қандай да бір факторға байланысты жүргізілуі мүмкін.

JavaScript браузер жұмысын бақылауға мүмкіндік береді. Мысалы, Window объекті экранға қалқып шығатын диалогтық терезелерді шығаруға, браузер жаңа терезелерін құруға, ашуға және жабуға, қозғалту режимдерін беруге және терезелер өлшемдерін беруге және т.б. мүмкіндік береді.

JavaScript құжаттамалар ішіндегілермен бірге өзара әрекет етуге мүмкіндік береді. Объект Document және оның ішіндегі болатын объектілер бағдарламаларға HTML - құжаттамалары бөліктерін оқуға және кейде олармен әрекет етуге мүмкіндік береді. Мәтінің дәл өзін оқуға мүмкін болмайды, бірақ, мысалы, сол құжаттаманың ішіндегі гипермәтінді сілтиемелер тізімін алуға мүмкіндік бар.

1.4 JavaScript тілінің кітапханалары мен фреймворктары

1.4.1 React

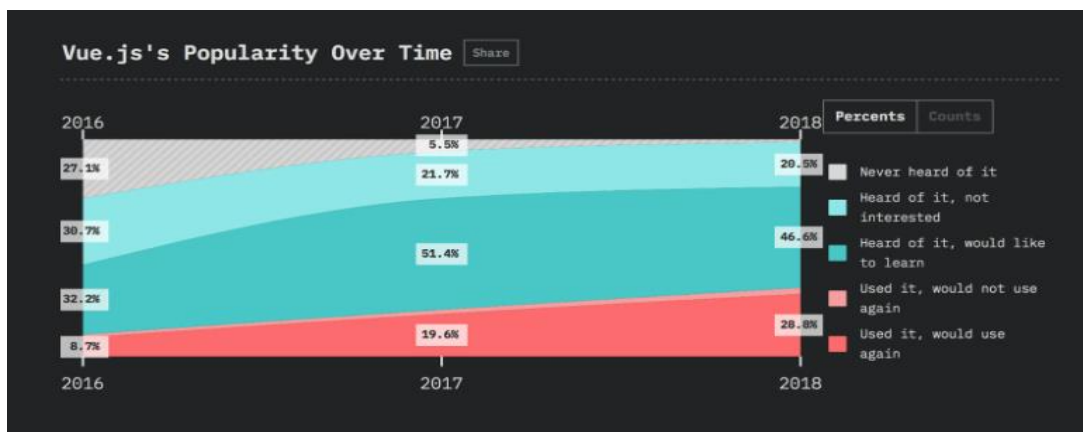
React (GitHub-та 110 мыңнан астам жұлдыз) – бұл Facebook командасымен UI құрастыруға арналған эффективті әрі икемді декларативті кітапхана. Ол ешқандай қиындықсыз интерактивті қолданушы интерфейсін құруға мүмкіндік береді. React объекті-центрлік қосымшаларды қолдап қана қоймай, оны жеңілдетеді. Сонымен қатар, фреймворкты құрастырушылар кері келісімділікті барынша ескерген болатын, сондықтан қосымшаныңыздың ұзақ өмір сүретіндігі туралы уайымдамасаныз да болады. Соңғы жылдары React кітапханасының деңгейі байқарлықтай өсті. React-программистер – javascript-программистер ішінде ең жоғары жалақы алушылар болып табылады.



1.5-сурет – React танымалдылығы

1.4.2 Vue

Vue.js – қолданушы интерфейсін жинаушы прогрессивті JavaScript-фреймворк. Басқа JavaScript кітапханаларымен жеңіл интеграцияланады. Веб-фреймворк ретінде реактивті стилде бір бетті қосымшалар құру функциясын атқара алады.



1.6-сурет – Vue.js танымалдылығы

1.4.3 Angular

Angular – Gitub-та 44 мың жұлдыз жинаған Google компаниясының фреймворкы. Өз алдына веб қосымшаларды құрауды жеңілдететін платформа. Angular-да декларативті шаблондар, тәуелділікті енгізу, екіжақтық байланыс және өңдеу барысындағы проблемаларды дұрыс жолмен шешу біріктіріледі. Бұл платформа веб қосымшаларды, мобильді қосымша және компьютерлерге құрауға мүмкіндік береді. Онда ең ыңғайлы және түсінікті командалық жолдың интерфейсі және консоль – графикалық интерфейспен клиент.

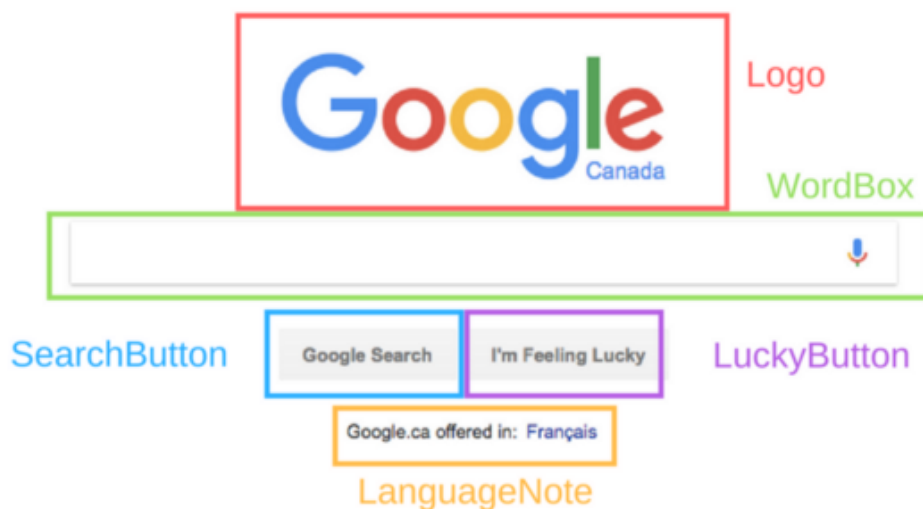


1.7-сурет – Angular танымалдылығы

1.5 Веб-сайт құру барысында пайдаланатын құралдар мен жабдықтар

Веб-сайт құру үшін бізге ең алдымен пайдалатын құралдар міндетті түрде қажет болады. Олар: HTML, CSS, JavaScript және серверлік кез-келген тіл (PHP, Python, Java және т.б.). Бірақ бұл құралдарды ғана пайдалану бізге жеткіліксіз. Себебі қазір заманауи технологиялардың дамуына байланысты аталған құралдардың әрқайсысының өзіне әртүрлі фреймворктармен кітапханалар жетерлік. Сондықтан да жоғарыда айтылған фреймворктың бірі React-ты пайдаланамын. React – пайдаланудың себебі және артықшылығы ол компоненттерге бөліп жұмыс істеуге мүмкіндік береді және сайтты ыңғайландырады, функционалдылық береді, сайтты барынша оңтайландырады. Мысалы, веб-бетте белгілі бір жерге өзгерту енгізу үшін барлық кодты емес тек бір компонентке өзгерту енгізесін. Бұл тек программистке ғана емес қолданушыларға барынша ыңғайлылық береді.

React – пайдаланушы интерфейстерін жасау құралы. Оның негізгі міндеті – веб-беттерде көруге болатын нәрселерді көрсету. Әрбір бетті кішігірім бөліктерге бөлу арқылы React интерфейстерді жасауды жеңілдетеді. Біз осы үзінділерді компонент деп атаймыз.



1.8-сурет – Бетті компоненттерге бөлу үлгісі

Суретте көрсетілген беттің әрбір таңдалған бөлігі, компонент болып саналады.

React компоненті, егер қарапайым сөзбен айтқанда, веб-беттің бір бөлігін білдіретін код бөлігі. Әрбір компонент - фрагментті білдіретін код бөлігін қайтаратын JavaScript функциясы. Бетті қалыптастыру үшін біз осы

функцияларды белгілі бір тәртіппен шақырамыз, шақырулардың нәтижелерін бірге жинап, оларды пайдаланушыға көрсетеміз.

React – JSX деп аталатын бағдарламалау тілін қолданады, ол HTML-ге ұқсас, бірақ оның html-ден ерекшелігі JavaScript ішінде жұмыс істейді. Бір бетті және мобильді қосымшаларды әзірлеу үшін react-ты пайдалануға болады. Оның мақсаты жоғары жылдамдықты, қарапайымдылықты және масштабтауды қамтамасыз ету болып табылады. Пайдаланушы интерфейстерін дамыту үшін кітапхана ретінде, React басқа Redux сияқты басқа кітапханалармен жиі пайдаланылады. React виртуалды DOM пайдаланады. React браузер DOM оңтайлы жаңарту үшін алдыңғы және ағымдағы интерфейс жайларының арасындағы айырмашылықты есептеуге мүмкіндік беретін жадыдағы кэш құрылымын жасайды. Осылайша, программист беттің барлығымен жаңартылғанын ескере отырып, кез-келген бетте жұмыс істей алады, бірақ кітапхана беттің қандай компоненттерін жаңартуы туралы өзі шешім қабылдайды.

JavaScript XML (JSX) интерфейстің құрылымын сипаттау үшін HTML-ға ұқсас синтаксисті пайдалануға мүмкіндік беретін JavaScript синтаксис кеңейтілімі. Әдетте, компоненттер JSX көмегімен жазылған, бірақ әдеттегі JavaScript-ды пайдалануға да мүмкіншілік береді.

Программа жазу барысында бізге міндетті түрде жұмысты жеңілдету үшін ыңғайлы редактор немесе IDE қажет болады. IDE немесе мәтіндік редактордың мыңдаған түрі бар. Қандай мәтіндік редакторды пайдалансаңда ешқандай айырмашылық болмайды. Мәселе тек ыңғайлылықта. Сондықтан да мен барлық мәтіндік редактор мен IDE-лерді салыстыра отырып өзіме ыңғайлысы, яғни React және JS кодтарын жазғанға ыңғайлы болуы үшін Atom мәтіндік редакторын таңдадым.

Atom (бұрынғы Atomicity) – Node.js-те жазылған плагиндерді қолдаумен және Git басқару мүмкіншілігімен Linux, MacOS, Windows операциялық жүйелеріне арналған ашық кодты тегін мәтіндік редактор. Және проект жасау барысында Git таратылған нұсқаны басқару жүйесін пайдаланамын. Себебі қазіргі таңда Git пайдаланбау сауатсыздықпен бірдей болып есептелінеді. Atom мәтіндік редакторында Git – пен жұмыс жасау мүмкіншілігі бар. Бұл редакторды таңдауымның тағы бір себептерінің бірі сол.

Git (айтыл. "гит") – үлестірілген нұсқаларды басқару жүйесі. Жобаны Линус Торвалдс Linux өзегін басқару үшін енгізген, бірінші нұсқасы 2005 жылдың 7 сәуірінде шықты. Бүгінгі күні Джунио Хаманодан (ағылш. *Junio C. Hamano*) қолдау алады. Git пайдаланатын жобалар мысалдарына Linux ядросын, Swift, Android, Drupal, Cairo, GNU Core Utilities, Mesa, Wine, Chromium, Compiz Fusion, FlightGear, jQuery, PHP, NASM, MediaWiki, DokuWiki, Qt және кейбір Linux дистрибутивтері. Жүйе бағдарламалар жиынтығы ретінде жобаланған, оларды скрипттерде пайдаланылуын ескере отырып арнайы әзірленген. Бұл Git негізінде ыңғайлы мамандандырылған нұсқаларды бақылау жүйесін немесе пайдаланушылық интерфейстерді жасауға мүмкіндік береді. Мысалы, Cogito дәл осындай - Git репозиторийлеріне үлгі қабық болып

табылады, ал StGit түзетулер коллекцияларын (патчтарды) басқару үшін Git пайдаланады. Git жылдам бөлу және нұсқаларын біріктіруді қолдайды, визуалдауға арналған құралдар және бейсызық әзірлеу тарихын шарлауды қамтиды. Darcs, BitKeeper, Mercurial, Bazaar және Monotone сияқты, Git әрбір әзірлеушіге әзірлеудің бүкіл тарихының жергілікті көшірмені ұсынады, өзгертулер бір репозиторийден басқасына көшірмесі жасалады. Git репозиторийлеріне қашықтан қол жеткізу git-daemon, SSH немесе HTTP-сервер арқылы қамтамасыз етіледі. git-daemon TCP-сервисі Git дистрибутивіне кіреді және SSH-бен қатар, ең көп тараған және сенімді қол жеткізу әдісі болып табылады. HTTP әдісі бойынша қатынау бірқатар шектеулерге қарамастан, бақыланатын желілерде өте танымал болып табылады, өйткені қолда бар желілік сүзгілер конфигурациясын пайдалануға мүмкіндік береді. GitHub – IT жобалардың хостингына және олардың бірлескен құрастыруларына арналған үлкен веб-сервис болып табылады. Веб-сервис нұсқаларды басқару Git жүйесінде құралған. Сервис ашық кодты проектiлерге тегiн. Сервистiң ұраны – “Social coding” – аударғанда “Бiрге код жазайық”.

2 Жобалау бөлімі

2.1 UI әзірлеу әдістері мен құралдары

Интерфейс кез-келген бағдарламалық жасақтаманың жүйесі үшін маңызды және оның негізінен соңғы пайдаланушыға бағытталған ажырамас бөлігі болып табылады. Бұл пайдаланушы бағдарламаны тұтастай бағдарламаға бағындыратын интерфейс арқылы, сонымен қатар, пайдаланушы жиі пайдаланушының интерфейсі қаншалықты ыңғайлы және түсінікті екеніне қарай қолданбалы бағдарламаны пайдалану туралы шешім қабылдайды. Алайда интерфейсстің дизайны мен дамуының күрделілігі өте үлкен. Сарапшылардың бағалауы бойынша, орташа есеппен жобаның жартысынан көбі. Бағдарламалық қамтамасыз ету жүйелерін дамытуға және қолдау көрсетуге жұмсалатын шығындардың төмендеуі немесе өнімділікті дамытудың қарапайымдылығы, қызмет көрсетудің қарапайымдылығы және пайдаланушы тәжірибесі бар тиімді бағдарламалық құралдарды әзірлеу маңызды. Бағдарламалық қамтамасыз ету жүйелерін әзірлеу мен қолдаудың өзіндік құнын төмендетудің бір жолы - аталған бағдарламалық жасақтаманың жоғары деңгейде сипатталуына мүмкіндік беретін құрал-жабдықтардағы төртінші буындағы құралдардың болуы, содан кейін спецификацияға сәйкес орындалатын кодты автоматты түрде генерациялау. Бағдарламалық қамтамасыз ету нарығы оның дамуына арналған кең ауқымды құралдарды ұсынады. Дегенмен, қол жетімді құралдар төртінші буын арқылы пайдаланушы интерфейсіннің кейбір құрамдастарының ғана дамуын қолдайды, ал қалған компоненттерін әзірлеуші әзірлейді, бұл шығындарды, дамудың күрделілігін және техникалық қызмет көрсетуді қиындатады. Пайдаланушылық интерфейссті дамытуға байланысты зерттеулер арнайы тілдесу бағдарламаларының тілдерінде пайда болғаннан бастап басталды және қазіргі уақытта мамандандырылған интерфейссті дамыту құралдарының пайда болуына әкелді. Әдебиетте пайдаланушы интерфейсін әзірлеуге арналған құралдардың бірыңғай қабылданған жіктелімі жоқ. Осылайша пайдаланушылық интерфейссті дамытуға арналған бағдарламалық қамтамасыз ету екі негізгі топқа бөлінеді, пайдаланушылық интерфейссті (toolkits) және жоғары деңгейлі интерфейссті дамыту құралдарын (higher-level development tools). Пайдаланушы интерфейсін әзірлеуге арналған құралдар, әдетте, интерфейсстің компоненттерінің (мәзір, түймелер, айналдыру жолақтары және т.б.) примитивтерінің кітапханасын қамтиды және бағдарламашылармен пайдалануға арналған. Интерфейссті дамытудың жоғары деңгейлі құралдарын программалық емес бағдарламалар қолдануға болады және I / O функцияларын көрсетуге мүмкіндік беретін тілмен жабдықталған, сондай-ақ тікелей өңдеу әдістерін пайдаланатын интерфейс элементтерін анықтауға болады. Бұл құралдар диалог құрастырушыларының (interface builders) және UIMS - Пайдаланушы интерфейсін басқару жүйелерінің (User Interface Management Systems – UIMS) авторларын қамтиды. UIMS-ге қоса, кейбір авторлар User Interface Development Systems (UIDS) – пайдаланушылық

интерфейсті дамыту жүйелерін, User Interface Design Environment (UIDE) – пайдаланушылық интерфейсті дамыту ортасын және т.б. пайдаланады. Мамандандырылған интерфейсті жасау құралдары сізге пайдаланушы интерфейсінің дамуын жеңілдетуге мүмкіндік береді, ол әзірлеушіні техникалық сипаттамалармен пайдаланушылық интерфейс компоненттерін көрсету үшін шақырады. Интерфейсті белгілеудің бірнеше негізгі жолы бар. Олар:

- интерфейстік синтаксисті (декларативті, объектілі-бағдарланған, оқиғалар тілдері және т.б.) көрсету үшін арнайы тілдер пайдаланылатын тіл;
- графикалық ерекшеліктер, әдетте, визуалды бағдарламалау, бағдарламалау демонстрациялары мен мысалдар арқылы интерфейсті анықтаумен байланысты. Бұл әдіс шектеулі интерфейстер класын қолдайды;
- нысанға бағдарланған көзқарасқа негізделген интерфейстің сипаттамасы тікелей манипуляция деп аталатын принциппен байланысты. Оның басты қасиеті - тұтас алғанда бүкіл жүйеге емес, жеке объектілермен өзара әрекеттесу. Нысандарды және басқарушы функцияларды басқару үшін қолданылатын типтік құрамдас бөліктер-өңдеушілер, мәзірлер, диалогтық алаңдар, әртүрлі түймелер;
- қолданба сипаттамалары үшін интерфейс сипаттамасы. Мұнда интерфейс қолданбалы міндеттің семантикасын ерекшелеуге сәйкес автоматты түрде жасалады. Алайда интерфейс сипаттамасының күрделілігі осы тәсілді іске асыратын жүйелердің тез пайда болу мүмкіндігін қиындатады.

2.2 UML бірыңғай модельдеу тілі

Программаға қосымша визуалды және құжаттамалық қолдау жасау – бұл көп уақытты талап ететін және қажырлы процесс. Көп уақыт кетеді және бағдарламалық жасақтама сәулеті қарапайым немесе анықтамалық болса, толықтай қажет болмайды. Алайда іс жүзінде программисттер әрдайым осындай міндеттерге тап болмайды. Көптеген жағдайларда бағдарламалық жасақтаманы әзірлеу кезінде, егер жүйе түзетулерді талап етсе, программисттер кодты қабылдайды және қателерді қалағандай түзейді, содан кейін нәтижені тұтынушыға көрсетеді. Көбінесе, шешім архитектурасы ауызша түсіндіріледі немесе қарапайым блок диаграммаларын қолданады. Осыған байланысты, Гради Буча әдісі (Booch), Джима Румбаха әдісі (OMT) және Айвара Джекобсона әдісі (OOSE) сияқты бұрынғы бірнеше стандарттарға негізделген әмбебап модельдеу тілі (UML) болды. Және оған үміт артты. Адамдар UML-мен жұмыс істеуге тырысты, бұл «күміс оқ атасы» болар деп үміттенді, бірақ ол кеңінен танымал болды. Зерттеушілер мемлекеттік диаграммалардың жалпы алгоритмдерді сипаттайтын және күрделі бағдарлама мінез-құлқын сипаттайтын жалпы

кабылданған құрал ретінде жаппай таралуына кедергі келтіретін үш негізгі кедергіні анықтайды. Біріншіден, күй диаграммалардан басқа мінез-құлықты сипаттау үшін диаграммалардың басқа түрлерін қолдану ұсынылды, бірақ олардың өзара әрекеттесуін реттейтін ережелер реттелмеген. Екіншіден, бағдарламалар құрылымы мен мінез-құлық сипаттайтын диаграммаларды ортақ пайдалану үшін ешқандай тәсілдеме ұсынылмады. Үшіншіден, мінез-құлықты сипаттайтын диаграммалар көбінесе әзірлеушілер тарапынан бір-бірімен байланысу үшін пайдаланылды, ал UML-тің мақсаты - ерекшелікті жасау және оны бағдарлама кодына аудару. UML - бағдарламалық жасақтама әзірлеу, бизнес-үдерістерді модельдеу, жүйелік жобалау және ұйымдастыру құрылымдарын көрсету үшін объектілі үлгілеу үшін графикалық сипаттама тілі. Барлық UML диаграммаларын екі топқа бөлуге болады, олардың біріншісі жалпы диаграммалар. Жалпы схемалар модельдеу тақырыбынан іс жүзінде тәуелсіз және кез-келген бағдарламалық жасақтамада субъектінің аймағына, шешім аймағына және т.б. UML мына диаграммалардан тұрады:

- құрылымдық диаграммалар;
- мінез-құлық диаграммалар;
- өзара әрекеттесу диаграммалары.

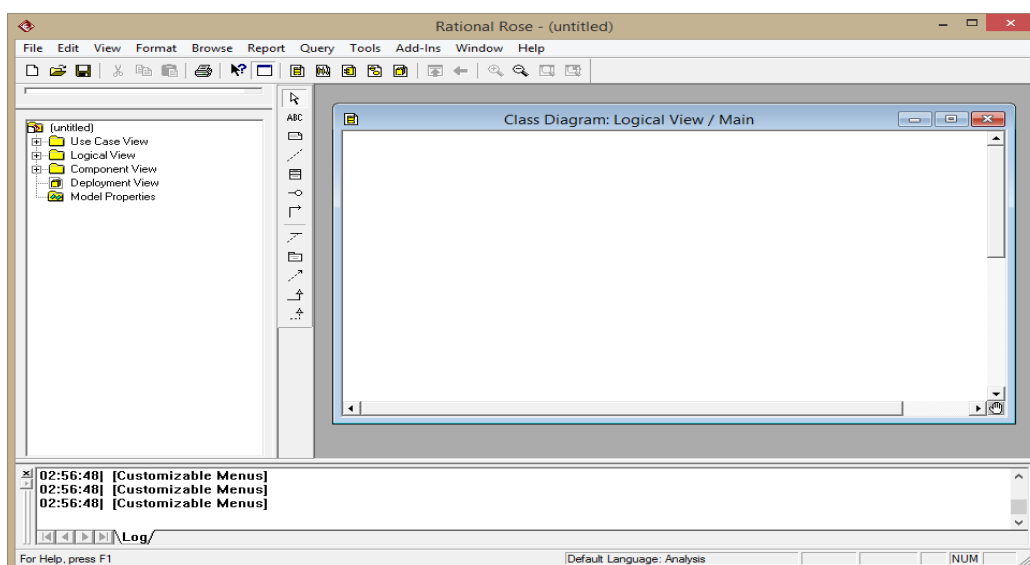
2.3 Rational Rose құралы

Rational Rose – ақпараттық жүйелерді және кәсіпорындарды басқарудың бағдарламалық жасақтамасын жобалау және дамыту үшін CASE құралы. Басқа CASE құралдары сияқты (ARIS, BPwin, ERwin) бизнес процестерін талдау және модельдеу үшін пайдаланылуы мүмкін. Бұл өнімнің бірінші нұсқасы Rational Software компаниясы шығарған. Кейінірек Rational Rose-ді IBM компаниясы сатып алды. Rational Rose-тың басқа құралдар арасындағы негізгі айырмашылық объектілі-бағытталған тәсіл болып табылады. Бұл құралды пайдалану арқылы жасалған графикалық модельдер объектілі-бағытталған принциптерге және UML (Unified Modeling Language) негізделеді. Rational Rose модельдеу құралдары әзірлеушілерге әртүрлі иерархия деңгейлері арасындағы барлық байланыстарды және бақылау әрекеттерін сақтап, толық кәсіпорын процесінің архитектурасын құруға мүмкіндік береді. Rational Rose-те бизнес-процестерді модельдеу түрлі аспектілерді қолдану арқылы жүзеге асырылады. Осы аспектілердің әрқайсысы процестердің нақты сипаттамалары мен мүмкіндіктеріне назар аударады.

CASE құралының басты артықшылығы модельдеудің объектілі принципімен байланысты. Оны қолданудың арқасында бизнес-процестерді модельдеуді жүзеге асыратын және модельдермен жұмыс істейтін түрлі мамандардың көзқарастарын мүмкіндігінше жақындастыруға болады. Сонымен қатар, Rational Rose бағдарламалық пакетімен жұмыс істеу ыңғайлылығына байланысты артықшылықтар бар. Rational Rose артықшылықтары олар:

- топтық қолдау;
- модельдерді басқару;
- қатені бақылау;
- модельдік құжаттама;
- конфигурацияны басқару.

Кез келген класстың ақпараттық жүйелерін көрнекі дамыту үшін қарапайым және қуатты шешім ретінде, Rational Rose үлгі жасаудың дұрыстығын жасауға, өзгертуге және тексеруге мүмкіндік береді. Rational Rose бағдарламалық жасақтаманың сәулетін сипаттау үшін стандартты графикалық таңбаларды анықтайтын UML әмбебап модельдеу тіліне негізделген әзірлеушілер тобын біріктіреді. Кез келген жоба қатысушылары - аналитиктер, модельдік сарапшылар, әзірлеушілер және басқалар - соңғы өнімді тиімдірек ету үшін Rational Rose-да салынған үлгілерді пайдалана алады. Веб программисттер мен XML программисттер үшін - Rational Rose - UML үшін веб-қосымшаларды кеңейтуді, JSP және ASP файлдарынан семантиканы кері бағдарламалауды, UML үшін веб-қосымшаларды кеңейтуді автоматтандыруды, Интернеттің архитектурасын нақты визуализациялауды қамтамасыз ететін жалғыз шешім. күрделі веб-сайттар және жақсартылған XML қолдау.



2.1-сурет – Программаның негізгі терезесі

2.4 Прецеденттер диаграммасы

Прецеденттер диаграммасы — актерлер мен қолданушылар арасындағы қатынастарды бейнелейтін диаграмма және жүйені тұжырымдамалық деңгейде сипаттауға мүмкіндік беретін пайдалану үлгісінің ажырамас бөлігі. Прецедент — бұл пайдаланушыға нақты, өлшенетін және қалаған нәтиже ала алатын модельдік жүйенің (оның функционалдық бөлігінің) мүмкіндігі. Прецедент жекелеген

сервистік жүйеге сәйкес келеді, оның қолданылу нұсқаларының бірін анықтайды және жүйемен өзара әрекеттесудің типтік әдісін сипаттайды. Прецеденттер диаграммасы әдетте сыртқы жүйе талаптарын анықтау үшін қолданылады. Диаграмманың басты мақсаты - тапсырыс берушіге, соңғы пайдаланушысына және әзірлеушіге жобаланған немесе бар жүйесін бірлесіп талқылауға мүмкіндік беретін функционалдылық пен мінез-құлықты сипаттау. Жүйені precedentтер диаграммасы арқылы модельдеу кезінде жүйе талдаушысы келесі жағдайларға ұмтылады:

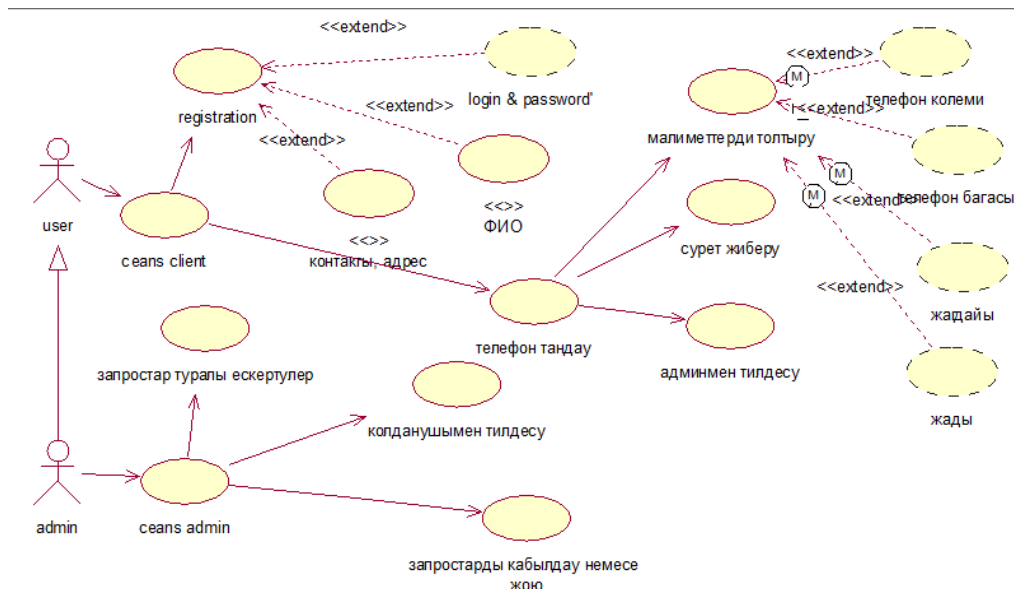
- жүйені оның ортасынан нақты бөлуге;
- актерлерді, олардың жүйемен өзара әрекеттесуін және жүйенің күтілетін функционалын анықтау;
- жүйенің функционалдығын толық сипаттаумен байланысты тақырыптық кеңістіктің тұжырымдамасын анықтау.

Прецеденттер диаграммасы диаграммада қолданыстағы топтар мен процеске қатысатын актерлер арасындағы қатынастар мен тәуелділіктер сипатталады. Қолдану диаграммалары жобаны көрсетуге арналмаған және жүйенің ішкі құрылымын сипаттай алмайтынын түсіну маңызды. Қолдану диаграммалары жүйенің болашақ пайдаланушылары, тұтынушылары бар өзара әрекеттесуді жеңілдетуге арналған және әсіресе жүйенің қажетті сипаттамаларын анықтау үшін пайдалы. Басқаша айтқанда, пайдаланудағы диаграммалар жүйеде қолданылған әдістерді көрсетпей-ақ не істеу керектігін көрсетеді. Прецеденттер диаграммасы актердің көзқарасынан жүйеде белгілі бір нәтижеге әкелетін әрекеттер тобын сипаттайды.

Прецеденттер диаграммасы жүйенің пайдаланушылары мен жүйенің өзі арасындағы типтік өзара әрекеттесу сипаттамалары болып табылады. Олар жүйенің сыртқы интерфейсін көрсетеді және жүйенің не істеу керектігін көрсетеді.

Актер – бұл жүйемен бірге пайдалану жағдайында өзара әрекеттесетін сыртқы көз (жүйенің элементі емес). Актерлер нақты адамдар (мысалы, жүйенің пайдаланушылары) және басқа компьютерлік жүйелер немесе сыртқы оқиғалар болуы мүмкін. Актерлер физикалық тұлғалар мен жүйелерді білдірмейді, бірақ олардың рөлі. Бұл адам әртүрлі тәсілдермен жүйеде (әртүрлі рөлдерді ескере отырып) өзара қарым-қатынаста болғанда, оны бірнеше актерлер көрсетеді. Мысалы, қолдау қызметінде жұмыс істейтін және клиенттерден тапсырыстарды қабылдаған адам жүйеде «қолдау бөлімінің мүшесі» және «сату бөлімінің мүшесі» ретінде көрсетіледі. Актерлерде пайдалану жағдайларымен байланыстың екі түрі болуы мүмкін. Олар:

- қарапайым ассоциация актер мен use case арасындағы сызықпен көрсетіледі (көрсеткі жоқ). Актердің қарым-қатынасын көрсетеді және оны пайдалану;
- қатынасты біріктіру қарапайым ассоциациямен бірдей, бірақ ол қолданылу жағдайын актермен инициализациялайтынын көрсетеді. Көрсеткі көрсетілген.



2.2-сурет – Прецедент диаграммасы

2.5 Тізбек диаграммасы

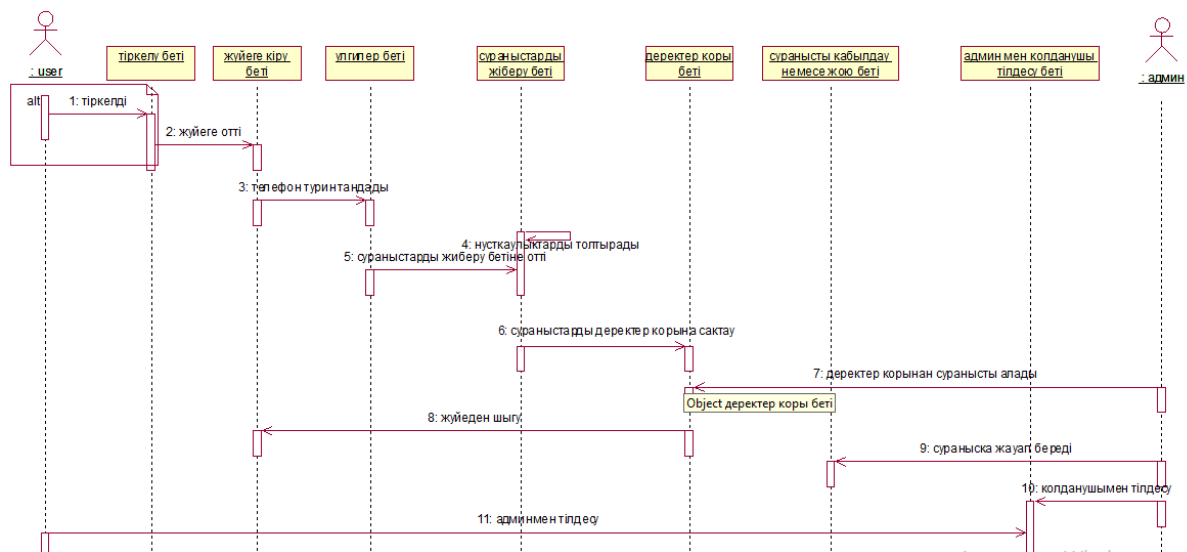
Тізбек диаграммасы (sequence diagram) – бір уақыттық осьтердегі белгілі бір объектілер жиынтығы, белгілі бір объектінің өмірлік циклі (белгілі бір заңды тұлғаны жою-жасау қызметі) және АЖ субъектілерінің (актерлерінің) өзара іс-қимылы шеңберінде диаграмма, немесе белгілі бір прецедент (сұрау жіберу және жауап алу). UML тілінің диаграммаларының бірі болып табылады. Тізбек диаграммасының негізгі элементтері уақыттың өтуін көрсететін нысандардың белгілерін (объектілердің атаулары бар тікбұрышты), уақыттың өтуін көрсететін тік тіршілік сызықтарын (өмірбаяны), белгілі бір функцияны көрсететін (төртбұрыштарды нүктелік «өмір сызығына») және объектілер арасындағы сигналдар немесе хабар алмасуды көрсететін көрсеткілер. Диаграмманың бұл түрінде объектілер солдан оңға қарай жайғасады. Жоғарыда айтылғандай, актерлер арасындағы өзара әрекеттесуді жіберушіден (оның көрсеткісі жүретін) алушыға (солға бағытталған көрсеткі бағытталатын) басқаруды беретін арнайы бағыттауыштар көрсетіледі. Көрсеткілер сценарийдің барысын және талданған прецедент кезінде болған оқиғаларды көрсетеді. Көрсеткіштің 5 түрі бар:

- синхронды хабарлама – жіберуші актер прецедентте қандай да бір әрекетті жасауы керек қабылдаушы актерге бақылауды жібереді. Алушы орындаған әрекет актерге аяқталмайынша (тиісінше жауап хабарламасы қабылданбайды) жіберуші актер кез келген әрекеттерді орындау мүмкіндігін жоғалтады. Үстіңгі үшбұрышты көрсеткі ретінде графикалық түрде бейнеленген, оның соңында объектінің әрекетін

көрсететін тіктөртбұрыш, оның соңында жауап беру туралы хабарлама бар;

- жауап беру хабарламасы – бұл хабар синхрондық хабарламаға жауап болып табылады. Әдетте, бастапқы жіберуші актерге қайтарылған кез келген мән бар, ол бақылауды қайтарады;
- асинхрондық хабар – жіберуші актер прецедентте кейбір әрекеттерді жасауы керек қабылдаушы актерге бақылауды жібереді. Синхрондық хабардың негізгі айырмашылығы - бұл жіберуші актер басқа әрекеттерді орындау мүмкіндігін жоғалтпайды;
- жоғалған хабарлама – алушының хабарламасы жоқ (жіберуші бар, алушы жоқ).
- хабарлама жіберуші жоқ хабар болып табылады.

Көрсеткіштің соңғы екі түрі (өзара әрекеттесу) өте сирек қолданылады. Олар негізінен бұл жағдайда сыртқы жүйелермен қолданыстағы объектілердің өзара әрекеттесуін көрсету үшін қолданылады. Іс жүзінде тізбек диаграммасы - жүйенің белгілі бір сессиясының хаттамасының жазбасы. Объектілі-бағдарланған бағдарламалау кезінде іске қосу кезіндегі ең маңызды нәрсе өзара әрекеттесетін нысандар арасында хабарлар жіберу болып табылады. Бұл осы диаграммада көрсетілетін хабарламалардың тізбегі.



2.3-сурет – Тізбек диаграммасы

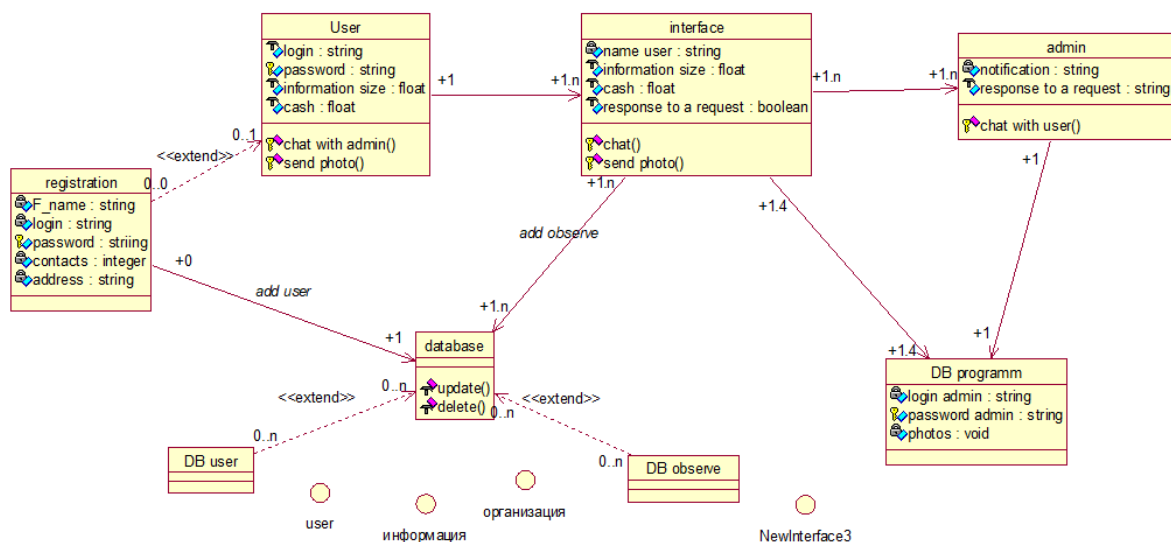
2.6 Класстар диаграммасы

Класстық диаграмма (class diagram) – бұл UML модельдеу тілінің құрылымдық схемасы, ол жүйелік сыныптардың иерархиясының жалпы құрылымын, олардың кооперацияларын, атрибуттарын (өрістерін), әдістерін, интерфейстерін және олардың арасындағы қарым-қатынасты көрсетеді. Ол

құжаттама мен визуализация үшін ғана емес, сондай-ақ тікелей немесе кері жобалау арқылы жобалау үшін кеңінен қолданылады. Класстардың диаграммасын құру мақсаты жүйенің декларативті элементтерінің (класстар, типтер және т.б.) статикалық құрылымын бейнелейді, сондай-ақ мінез-құлықтың кейбір элементтерін (мысалы, операциялар) қамтиды, бірақ олардың динамикасы диаграммалардың басқа түрлерінде (байланыс диаграммалары, мемлекеттік диаграммалар). Ыңғайлы болу үшін класс диаграммасы кірістірілген қоса пакеттің көрінісімен толықтырылуы мүмкін. Жаһандық объектілерді ұсынғанда, әзірлеуші олардың қазіргі жағдайын, мінез-құлқын және өзара қарым-қатынастарын көрсетуі керек. Әрбір кезеңде шындыққа (өнімділік, инкапсуляция, көріну және т.б.) қатысы жоқ маңызды мәліметтер мен тұжырымдардан абстракция жасалады. Сабақтар түрлі деңгейлер тұрғысынан қаралуы мүмкін. Әдетте, олар үш негізгі бағытпен ерекшеленеді: аналитикалық деңгей, жобалық деңгей және іске асыру деңгейі. Класс диаграммасы объектілі-бағытталған модельдеудегі негізгі элемент болып табылады. Класстар диаграммасы үш компоненттен тұратын шеңберлерде ұсынылған:

- жоғарғы бөлікте класс атауы жазылған. Класс атауы шоғырланған және қалың жазылған. Класс атаулары бас әріппен басталады. Егер класс дерексіз болса, онда оның аты қалың курсивпен жазылған;
- ортасында - класстың өрістері (атрибуттары). Олар солға теңестіріледі және кіші әріппен басталады;
- төменгі бөлігінде класстық әдістер бар. Олар сондай-ақ солға теңестіріледі және кіші әріппен жазылады.

Класстық диаграмма – жүйенің құрылымын сипаттаудың негізгі жолы. Бұл таңқаларлық емес, себебі UML ең алдымен объектілі-бағытталған тіл болып табылады, ал сабақтар – ең бастысы (егер жалғыз болса) «құрылыс материалы». Класқа арналған диаграммада субъектінің бір негізгі түрі пайдаланылады: кластар (кластардың көптеген ерекше жағдайлары: интерфейстер, қарабайыр түрлері, қауымдастығының кластары және басқалар), олардың арасында келесі негізгі қатынас түрлері белгіленеді.



2.4-сурет – Класс диаграммасы

Класс – бұл атрибуттармен, операциялармен, қатынастармен және семантикамен бірдей нысандар жиынтығы. Графикалық түрде сынып көлденең сызықтармен 3 блокқа бөлінген тіктөртбұрыш ретінде бейнеленген. Атрибуттар мен операциялар үшін үш түрі көрінеді:

- private (жеке);
- protected (қорғалған);
- public (ортақ).

Өрістер мен әдістердің көрінуі сол элементтің атауымен жолдағы сол таңба ретінде көрсетіледі. Әрбір класстың басқа класстардан ерекшеленетін атау болуы керек.

Осы технологиялар мен құралдарды пайдалана отырып, веб-сайттың жобасын құрастырдым. Және осы модельдерді құру кез-келген жобада міндетті болып табылады, әрі ыңғайлы.

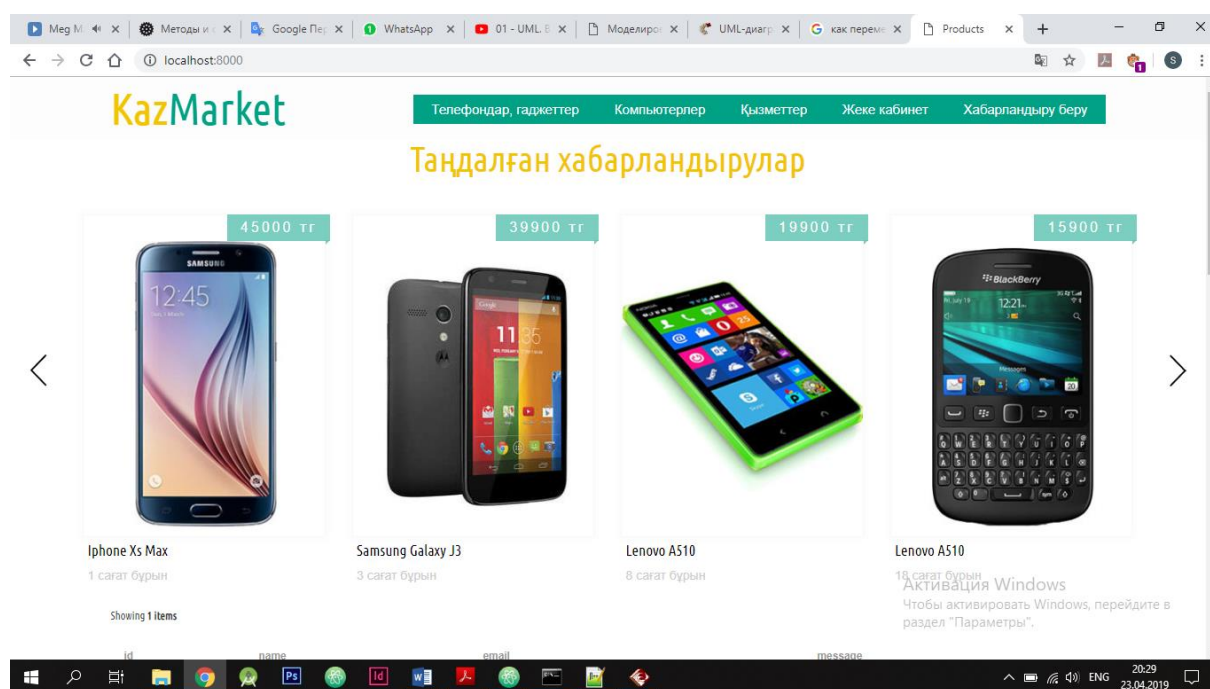
3 Қолданбалы бөлім

3.1 Ұялы телефондар мен компьютерлерге арналған маркеттің веб-сайты

Ұялы телефондар мен компьютерлерге арналған маркеттің веб-сайты барлық стандарттар мен талаптарға сәйкес блоктар мен веб беттерден тұрады. Сайттың әрбір беті заманауи дизайн талаптарына сай элементтерден тұрады. Және қолданушыға барынша ыңғайлы. Веб-сайттың негізгі беті төмендегілерді қамтиды:

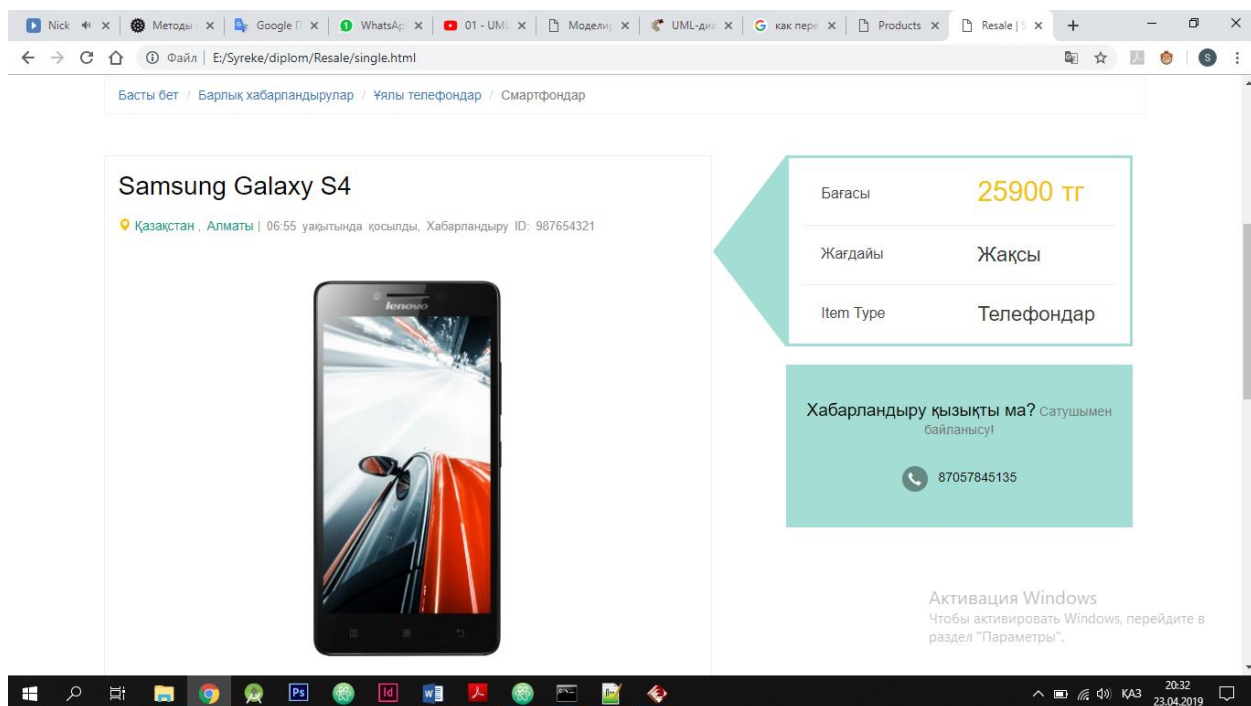
- сайттың шапкасы (header);
- слайдер;
- сайттың денесі (body);
- беттің төменгі бөлігі (footer).

Веб-сайттың структурасы кез-келген заманауи сайтқа сай. Адрес жолағына сайттың адресін енгізетін болсақ, ең бірінші болып сайттың негізгі беті ашылады. Бұл бет қолданушының ең алғаш көретін беті болып табылады. Ол 3.1-суретте көрсетілген.



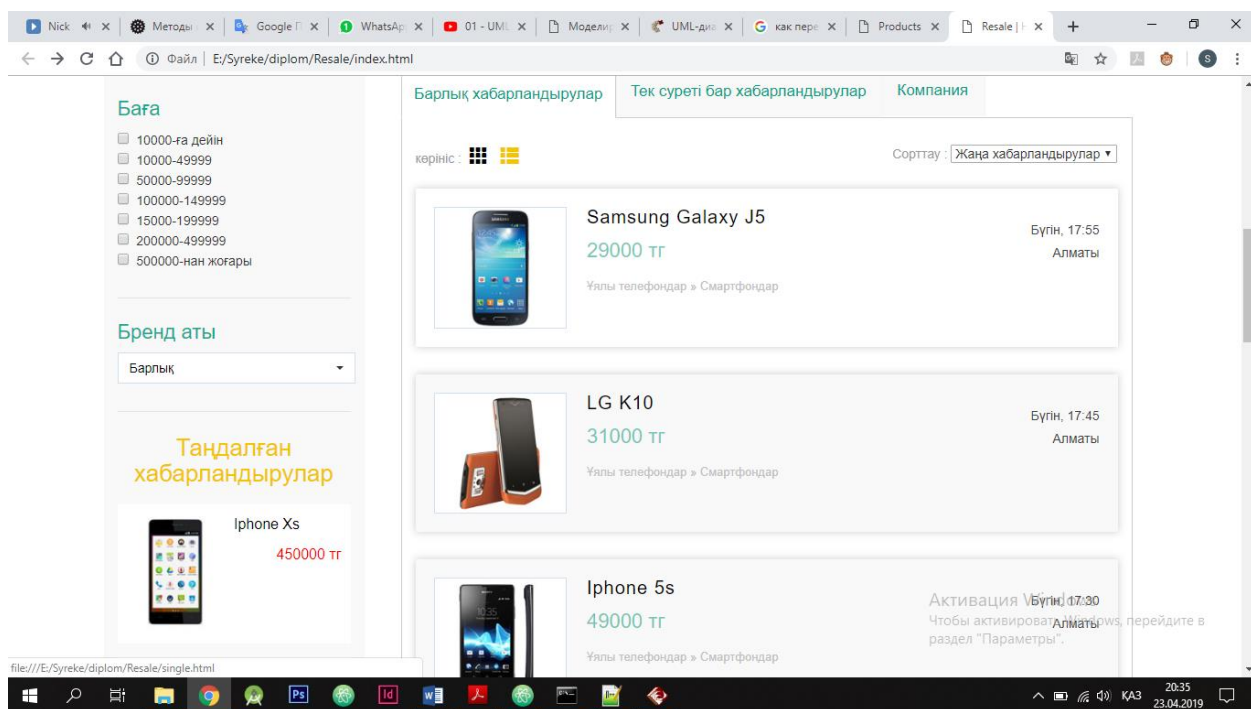
3.1-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының негізгі беті

Егерде слайдерде көрсетілген хабарландыру тізімінен белгілі бір тауарды басқанда арнайы тауар беті ашылады. Ол бет 3.2-суретте көрсетілген.



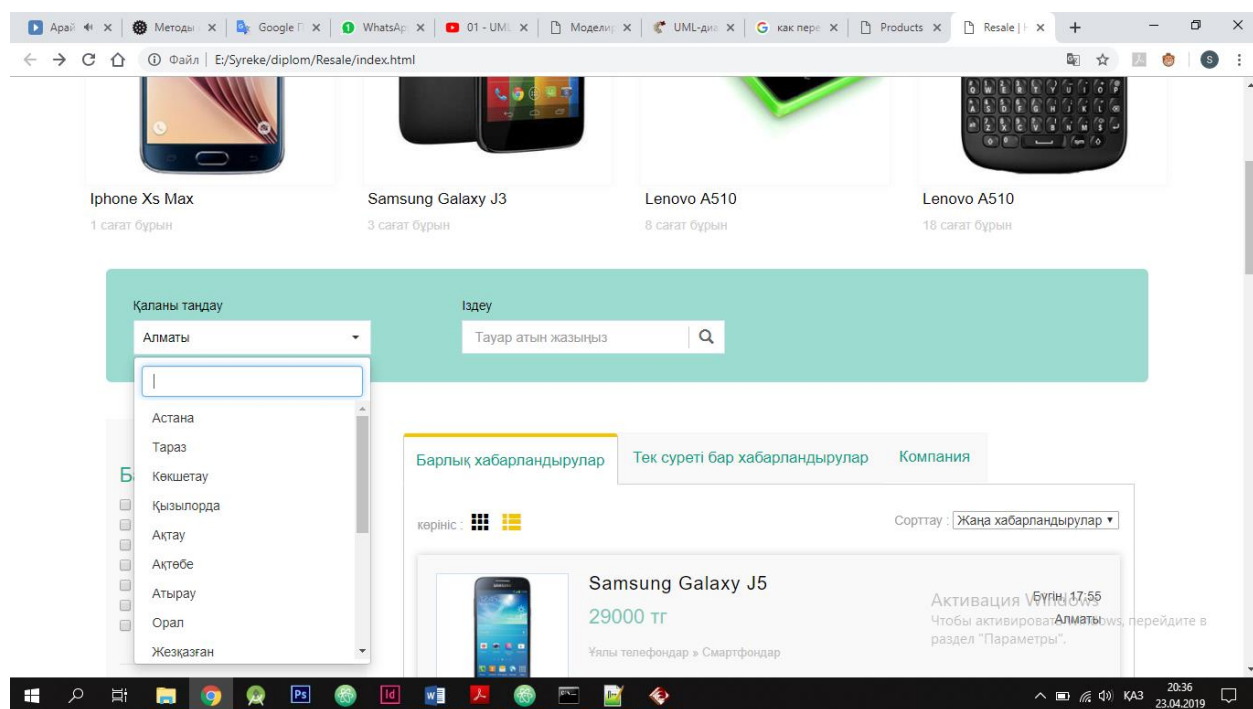
3.2-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының тауардың жеке беті

Ары қарай веб-сайтты төмен қарай жылжитсақ, басты хабарландыру тізімі көрінеді. Бұл жерде тізімдегі хабарландыруларды сұрыптауға, бренд аты бойынша қарауға болады. Бұл бет 3.3-суретте көрсетілген.



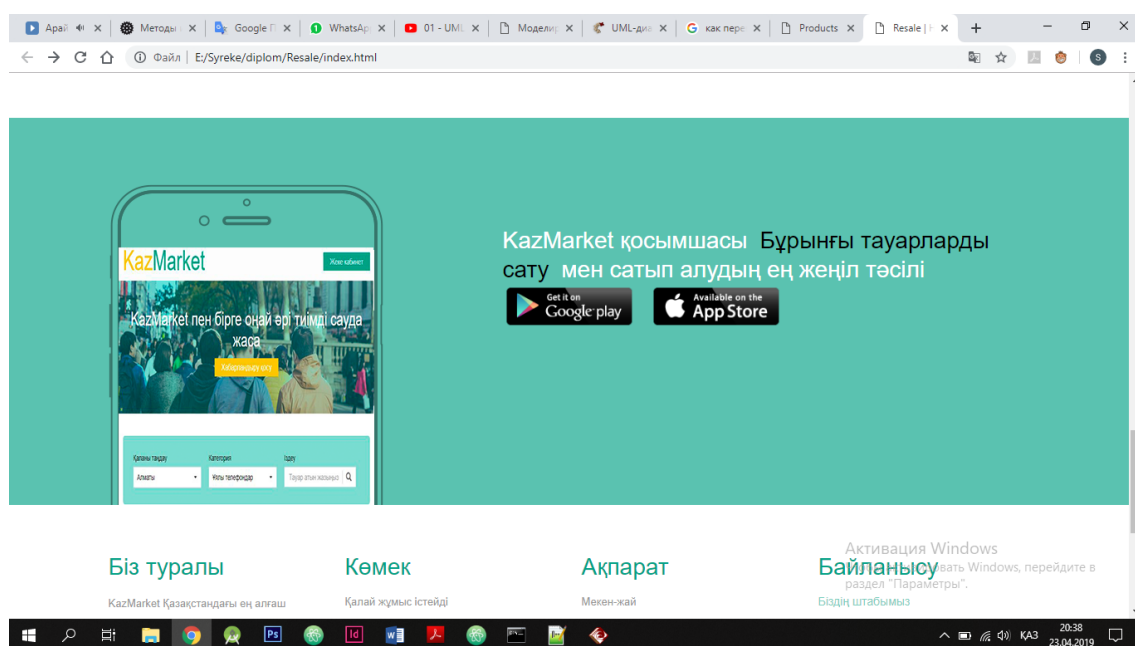
3.3-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының тауарлар беті

Сол хабарландырулар тізіліп орналасқан бетте аты бойынша іздеу және қаланы таңдау мүмкіншілігі бар. Ол 3.4-суретте көрсетілген.



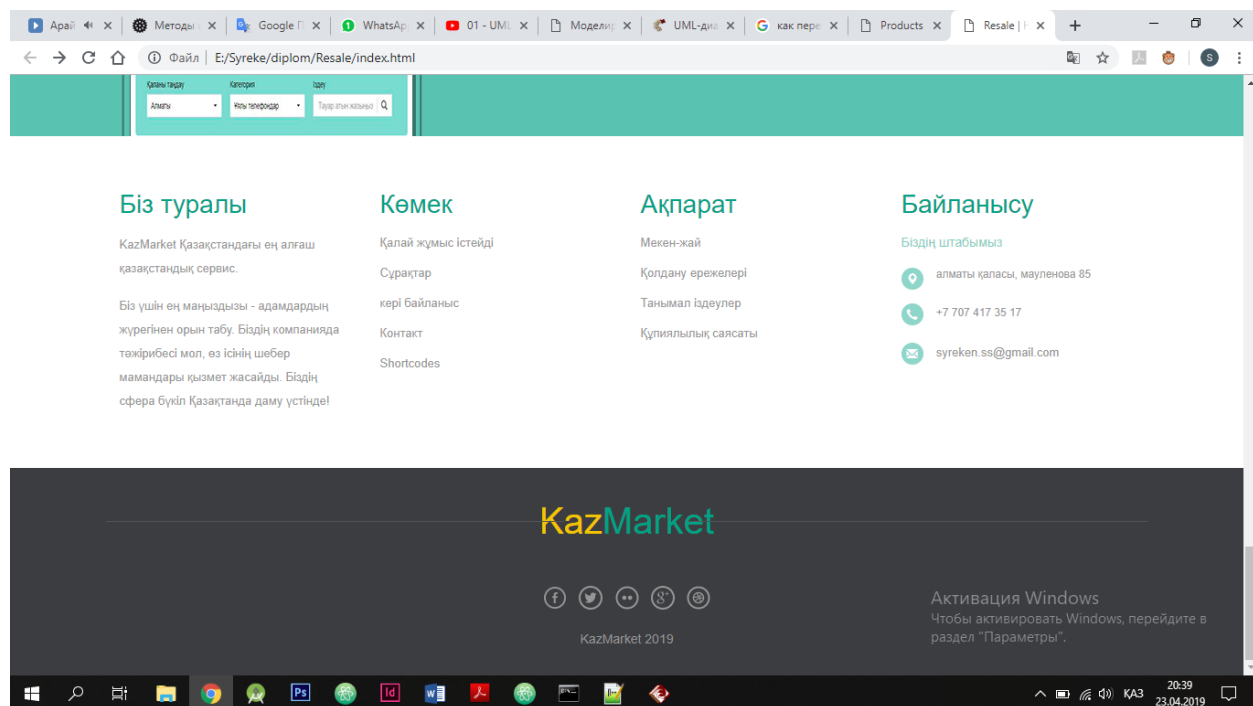
3.5-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының қала таңдау мен іздеу блогы

Ары қарай төмен жылжытқанда веб-сайттың жарнама блогы көрінеді. Ол 3.5-суретте көрсетілген.



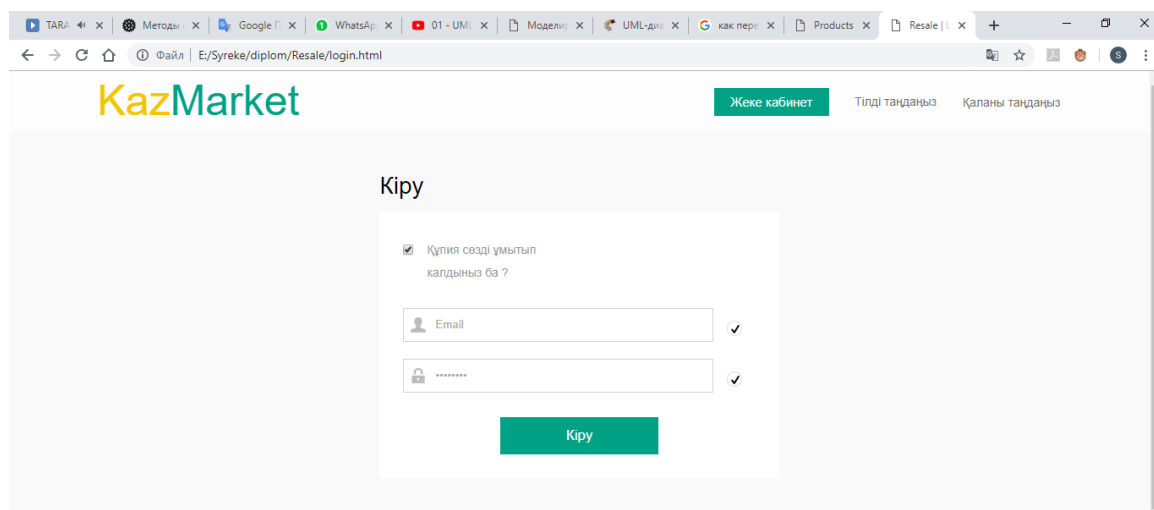
3.5-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының жарнама блогы

Сайттың төменгі бөлігінде “Біз туралы”, “Көмек”, “Ақпарат”, “Байланысу” навигациялары және әлеуметтік желілерге сілтемелер орналасқан. Ол 3.6-суретте көрсетілген.



3.6-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының төменгі деректемесі

Сайт бетінде барлық заманауи сайттардағыдай жоғарғы меню (navbar) орналасқан. Және сол жердегі «Жеке кабинет» немесе «Хабарландыру беру» батырмасын басқан жағдайда жеке кабинетке кіру беті ашылады. Ол 3.7-суретте көрсетілген.



3.8-сурет – «Ұялы телефондар мен компьютерлерге арналған маркет» веб-сайтының жеке кабинетіне кіру беті

ҚОРЫТЫНДЫ

Дипломдық жұмысты орындау барысында «Ұялы телефондар мен компьютерлерге арналған маркет» тақырыбына байланысты веб-сайт құрылды. Веб-сайт барлық заманауи технологияларды пайдалана отырып жасалды. Қолданушыларға барынша ыңғайлы болатын қолданушы интерфейсі. Түсінікті әрі функционалды дизайн. Тауарлардың барлық түрі, фильтр, іздеу, жеке кабинет барлығы ойластырылған.

Веб-сайтты жасау үшін барлық заманауи, қажетті және сайтты барынша оңтайландыратын технологиялар қолданды. Ол технологиялар сайттын жылдамдығын әлдеқайда арттырып, қолданушыларға еш қолайсыздық туғызбайды. Веб-сайт барлық стандарттарға сай жасалды.

«Ұялы телефондар мен компьютерлерге арналған маркет» сайтында қолданушылар өздеріне ұялы телефон (смартфон) және компьютерлерді салыстыра отырып, өздеріне ұнаған тауарды сатып ала алады. Және де өздерінің қолданып жүрген ұялы телефондарын немесе компьютерлерін еш қиындықсыз хабарландаруға қоса алады. Ол үшін жай ғана қолданушылардың қауіпсіздігі мен ыңғайлылығы үшін жасалған тіркелу формасын толтырса жеткілікті. Ары қарай қолданушылар өздерінің хабарландыруға берген тауарларын өздерінің жеке кабинетінен көріп, бақылай алады. Сонымен қатар адамдар егер ұялы телефон немесе компьютер іздесе басқа olx немесе market сияқты сервистер мен сайттарды пайдаланудың еш қажеттілігі жоқ.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Джон Дакетт; [пер.с англ. М. А. Райтмана]. HTML и CSS. Разработка и дизайн веб-сайтов. – М.: Эксмо, 2013. –480 с.
- 2 Современный учебник JavaScript // Сайттың электрондық нұсқасы <https://learn.javascript.ru/>
- 3 React – a JavaScript library for building user interfaces // Сайттың электрондық нұсқасы <https://reactjs.org/>
- 4 Методы и средства разработки пользовательского интерфейса // Сайттың электрондық нұсқасы <http://guimachine.ru/?p=937>
- 5 UML – диаграмма классов // Сайттың электрондық нұсқасы <https://prog-cpp.ru/uml-classes/>
- 6 О сайте – Kolesa.kz // Сайттың электрондық нұсқасы <https://kolesa.kz/content/about/>
- 7 Крыша. Недвижимость в Казахстане // Сайттың электрондық нұсқасы <https://krisha.kz/>
- 8 Буч Г., Рамбо Д., Якобсон И. Введение в UML от создателей языка. – М.: ДМК Пресс, 2015. – 496 с.
- 9 Рамбо Д., Блаха М. Объектно-ориентированное моделирование и разработка. – СПб.: Питер, 2007. – 544 с.
- 10 Трофимов С.А. CASE-технологии: практическая работа в Rational Rose. – М.: Бином-Пресс, 2002. – 288 с.
- 11 Леоненков А. Самаучитель UML. BHV – Санкт-Петербург, 2001. – 304 с.
- 12 Anthony Accomazzo, Ari Lerner, Nate Murray, Clay Allsopp, David Guttman, and Tyler McGinnis. Fullstack React. The Complete Guide to ReactJS and Friends - San Francisco, 2016-2019 Fullstack.io

А қосымшасы **(міндетті)**

Ұялы телефондар мен компьютерлерге арналған маркеттің клиенттік бөлігін құру

А.1 Кіріспе

Дипломдық жобаның негізгі идеясы ұялы телефондар мен компьютерлерді сатуға және сатып алуға қажетті оңтайлы жүйе құру.

Идеаны жүзеге асыру барысында заманауи технологиялар қолдану есепке алынды. Дипломдық жобаның мақсаты – ұялы телефон немесе компьютер сатып алушы адамдар мен сатушы адамдар арасындағы байланысты жеңілдету және сауданы оңтайландыруға арналған веб-сайт құру.

А.1.1 Өңдеудің мақсаты мен қызметі

Дипломдық жобаның мақсаты: Ұялы телефон немесе компьютер сатып алушы адамдар мен сатушы адамдар арасындағы байланысты жеңілдету және сауданы оңтайландыруға арналған веб-сайт құру.

Дипломдық жобаның өзектілігі – адамдардың жұмысын оңтайландыру, тапсырмаларды жоспарлауға мүмкіндік берілуі және олардың орындалуын бақылау. Клиенттерден келіп түскен тапсырмаларды уақытылы орындалуы. Нарықтың дамыған заманында компаниялар арасындағы бәсекелестік өте үлкен, сондықтан осы веб-сайт клиенттерді жоғалтпауға көмек береді.

Дипломдық жобаның есептік қойылымы

Құрылатын жүйеде кәсіпорын мен клиенттерге қатысты барлық ақпараттар деректер қорында сақталатын веб-сайт құру.

- а) веб-сайт құрылымын құру;
- б) бағдарламалау тілін таңдау;
- в) деректер қорын басқару жүйесін таңдау;
- г) тиімді және түсінікті программалық интерфейсті құру;
- д) қосымша бойынша ақпарат алуды ұйымдастыру;
- е) веб-сайт қолданушыларының логин пароль арқылы программаға кіруі.

Дипломдық жобаның нысаны

Қолданушыларының жұмысын жеңілдетуге арналған веб-сайт.

Дипломдық жобаның міндеттері

- а) құжаттар мен есептерді автоматтандыру;
- б) командада жұмысты ұйымдастыру;

А қосымшасының жалғасы

в) тапсырмалардың уақытылы орындалуын қадағалап, жұмыстың болмай қалмауын және сәтсіздікке ұшырамауына мүмкіндік бермеу.

А.1.2 Қолдану саласы

Бұл веб-сайт арқылы клиенттермен компаниялар арасындағы қарым-қатынас көрсетіледі және қызмет сапасын жоғарылатуға мүмкіндік береді.

А.1.3 Анықтамалар, терминдер және қысқартулар

Анықтамалар терминдер және қысқартулар А.1-кестеде көрсетілген

А.1-кесте – Анықтамалар, терминдер және қысқартулар

Терминдер немесе қысқартулар	Анықтамалар
UI	Қолданушы интерфейсі (User Interface)
JSX	JSX - JavaScript синтаксисін қосатын препроцессор. JSX жоқ React пайдалануыңызға болады, бірақ JSX react-ты неғұрлым талғампаз етеді.
Git	таратылған нұсқаны басқару жүйесі

А.2 Жалпы сипаттамасы

Құрылатын жүйе арқылы клиенттерден келіп түскен тапсырмаларды басшының жұмысшыларына уақытылы жіберіп және сол жұмыстарды қабылдап алу, және клиенттерге дайын жұмыстарды жіберіп отыру.

Дипломдық жұмыстың нәтижесінде келесі негізгі тапсырмалар орындалуы қажет:

- веб-сайттың құрылымын құру;
- тиімді және түсінікті веб-сайт интерфейсін құру;
- компания қызметкерлері мен клиенттер арасындағы әрекеттерін ұйымдастыру;
- қосымша бойынша ақпарат алуды ұйымдастыру.

А қосымшасының жалғасы

А.2.1 Пайдаланушы интерфейстер

Ақпараттарды алу үшін басты бетте қолданушы түрін таңдап авторизация жасау қажет.

Қолданушының екі түрі бар:

- басшы;
- қызметкер.

Басшы сәтті авторизация өткеннен кейін ол өзінің мәзірінде толық қолданушыларын және тек өз бөліміндегі қолданушыларды көре алады. Дәл осы басшыға клиенттерден тапсырмалар келіп түседі және сол тапсырмаларды басшы өз қолданушыларына белгілі уақытта тапсырылу керек деп жібереді, сол жұмыстарды қабылдап алады. Қолданушыға жаңалықтар беті тапсырмалар келіп сол жұмысты атқарады.

А.2.2 Аппараттық интерфейстер

Atom мен жұмыс жасау үшін:

- процессор Intel® Core i3 және жоғары немесе 4 ГГц аналогтық процессор AMD® технологиясы бойынша SSE2;
- 2 Гб жедел жады (8 Гб болуы ұсынылады);
- 2 Гб қатты диск жадында бос орын болу керек;
- видео картасы 512Мб- тан кем болмауы (1 Гб және одан көп болуы ұсынылады);
- тінтуір – PS/2 стандартты;
- пернетақта (орысша/қазақша);
- монитор (көрсету қабілеті) – 17''(1024x768).

А.2.3 Программалық интерфейстер

Бағдарламалық қамтамасыздандыруды құру:

- а) Microsoft Windows, Mac OS X, Linux операциялық жүйелері;
- б) Node JS;
- в) NPM;

А қосымшасының жалғасы

А.2.4 Адаптация бойынша талаптар

Программалық кешен Windows 7, Windows 8, Windows 10, Mac OS X, Linux платформаларында жұмыс істеуі тиіс, бұл немесе басқа платформаларды қолдануда қажетті программалық компоненттердің қайсысы орнатылуы керек

А.3 Өндеуге талаптар

А.3.1 Функциональдық талаптар

Қызметкер немесе басшы жеке кабинетіне кіру үшін өзінің логин мен құпия сөзін енгізу керек. Қосымшаға кіргеннен кейін өзіне қажетті ақпараттарды көре алады.

А.3.1.2 Функциональды талаптарды ұйымдастыру

Жүйе клиент-сервер архитектурасының бөлігі болып табылады. Клиент пен сервер арасындағы байланыс транспортты-желі деңгейінде TCP/IP хаттамасының көмегімен жүзеге асады. Ақпараттық алмасуды қолданбалы деңгейдегі HTTP хаттамасы және оның HTTPS кеңейтілімі қолданылады.

А.3.2.1 Өнімділігі

Қосымшаның өнімділігін арттыру үшін ұялы телефон немесе компьютер Wi-Fi желісінде немесе 3G, 4G интернетте болуы керек.

А.3.2.2 Қатынау және сенімділік

Жүйе жұмысы жұмыс уақытының аралығында тоқтап қалмауы тиіс:

- профилактикалық жұмысты белгілі бір уақытта жүргізу керек;
- авариялық жағдайларды болдыртпау.

Жүйе пайдаланушылардың сенімді тоқтаусыз жұмысын қамтамасыз етіп, олардың кез-келген ісіне тоқтаусыз қызмет түрін көрсетуі керек.

А қосымшасының жалғасы

А.3.2.3 Қатені өңдеу

Жүйедегі барлық қателер өңделуі тиіс және қателер туралы хабарлар пайдаланушылар қабылдай алатын түрде көрсетілуі керек.

Хабарлау жүйесінде пайдаланушылардың мүмкін емес әрекеттерін, ақпарат енгізудің қате үлгілерін іске асыру керек.

А.3.2.4 Интерфейсті талаптар

Қосымша жылдам ақпарат алу үшін қолданушыға түсінікті және ыңғайлы интерфейсі болуы қажет. Қосымшаның интерфейсі жалпы қабылданған стандарттардан ауытқымауы қажет.

А.3.2.5 Шектеулер

Қолданылатын платформалар: Кез-келген интернетке мүмкіндігі бар Android/IOS платформасындағы телефондарда немесе планшеттерде және компьютерлерде жұмыс істейді.

А.3.3 Кері талаптар

Процессор типі – Intel core i3 3 ГГц немесе жоғары. Осы талап қанағаттандырылмаса нәтижеге жету мүмкін емес.

Керісінше талаптар бұл талаптардың спецификациясы. Онда қосымша нені жасамауы жайлы анықтайды. Керісінше талаптардың шексіз саны бар болуы логикалы: мүмкін болатын түсінбеушілікті түсіндіретінін ғана белгілеу қажет.

Б қосымшасы
(міндетті)

Бағдарлама мәтіні

1. base.html файлы

```
{% load static %}
<!DOCTYPE html>
<html>
<head>
  <title>{% block title %}KazMarket{% endblock %}</title>
  <!-- temp file potom udalit-->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/bulma/0.6.2/css/bulma.min.css">
  <link rel="stylesheet" href="{% static "css/bootstrap.min.css" %}">
  <link rel="stylesheet" href="{% static "css/bootstrap-select.css" %}">
  <link href="{% static "css/style.css" %}" rel="stylesheet" type="text/css"
media="all" />
  <link rel="stylesheet" href="{% static "css/flexslider.css" %}" type="text/css"
media="screen" />
  <link rel="stylesheet" href="{% static "css/font-awesome.min.css" %}" />
  <!-- for-mobile-apps -->
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <!-- //for-mobile-apps -->
  <!--fonts
  <link href='//fonts.googleapis.com/css?family=Ubuntu+Condensed' rel='stylesheet'
type='text/css'>
  <link
href='//fonts.googleapis.com/css?family=Open+Sans:400,300,300italic,400italic,600,
600italic,700,700italic,800,800italic' rel='stylesheet' type='text/css'>
  --><!--//fonts-->
  <link href="https://fonts.googleapis.com/css?family=Ubuntu+Condensed"
rel="stylesheet">
  <!-- js -->
  <script type="text/javascript" src="{% static "js/jquery.min.js" %}"></script>
  <!-- js -->
  <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
  <script src="{% static "js/bootstrap.min.js" %}"></script>
  <script src="{% static "js/bootstrap-select.js" %}"></script>
  <script>
    $(document).ready(function () {
      var mySelect = $('#first-disabled2');
```

Б қосымшасының жалғасы

```
$('#special').on('click', function () {
    mySelect.find('option:selected').prop('disabled', true);
    mySelect.selectpicker('refresh');
});

$('#special2').on('click', function () {
    mySelect.find('option:disabled').prop('disabled', false);
    mySelect.selectpicker('refresh');
});

$('#basic2').selectpicker({
    liveSearch: true,
    maxOptions: 1
});
});
</script>
<script type="text/javascript" src="{% static 'js/jquery.leanModal.min.js'
%}"></script>
<link href="{% static 'css/jquery.uls.css' %}" rel="stylesheet"/>
<link href="{% static 'css/jquery.uls.grid.css' %}" rel="stylesheet"/>
<link href="{% static 'css/jquery.uls.lcd.css' %}" rel="stylesheet"/>
<!-- Source -->
<script src="{% static 'js/jquery.uls.data.js' %}"></script>
<script src="{% static 'js/jquery.uls.data.utils.js' %}"></script>
<script src="{% static 'js/jquery.uls.lcd.js' %}"></script>
<script src="{% static 'js/jquery.uls.languagefilter.js' %}"></script>
<script src="{% static 'js/jquery.uls.regionfilter.js' %}"></script>
<script src="{% static 'js/jquery.uls.core.js' %}"></script>
<script>
    $( document ).ready( function() {
        $( '.uls-trigger' ).uls( {
            onSelect : function( language ) {
                var languageName = $.uls.data.getAutonym( language );
                $( '.uls-trigger' ).text( languageName );
            },
            quickList: ['en', 'hi', 'he', 'ml', 'ta', 'fr'] //FIXME
        } );
    } );
</script>
</head>
<body>
<div class="header">
```

Б қосымшасының жалғасы

```
<div class="container">
  <div class="logo">
    <a href="/"><span>Kaz</span>Market</a>
  </div>
  <div class="header-right">
    <a class="account">Телефондар, гаджеттер</a>
    <a class="account">Компьютерлер</a>
    <a class="account">Қызметтер</a>
    <a class="account" {% if request.user.is_authenticated %} href="/post_add"
{% else %} href="{% url 'login' %}"
{% endif %}>Add advert</a>
    {% if request.user.is_authenticated %}
      <a class="account" href="{% url 'dashboard' %}">My profile</a>
    {% endif %}
    <span class="user">
      {% if request.user.is_authenticated %}
        <a class="account" href="{% url 'logout' %}">Logout</a>
      {% endif %}
    </span>
  </div>
</div>
</div>

<!-- content-starts-here -->

<div class="total-ads main-grid-border">
{% block content %}
{% endblock %}
</div>
<div class="content">

  <div class="trending-ads">
    <div class="container">
      <!-- slider -->

    </div>
    <!-- //slider -->
  </div>
  <div class="mobile-app">
    <div class="container">
      <div class="col-md-5 app-left">
```


Б қосымшасының жалғасы

```
<a href="mobileapp.html"></a>
</div>
<div class="col-md-7 app-right">
  <h3>KazMarket қосымшасы <span>Бұрынғы тауарларды сату</span>
    мен сатып алудың ең жеңіл тәсілі</h3>

  <div class="app-buttons">
    <div class="app-button">
      <a href="#"></a>
    </div>
    <div class="app-button">
      <a href="#"></a>
    </div>
    <div class="clearfix"> </div>
  </div>
</div>
<div class="clearfix"></div>
</div>
</div>
<!-- footer section start-->
<footer>
  <div class="footer-top">
    <div class="container">
      <div class="foo-grids">
        <div class="col-md-3 footer-grid">
          <h4 class="footer-head">Біз туралы</h4>
          <p>KazMarket Қазақстандағы ең алғаш қазақстандық сервис.</p>
          <p>Біз үшін ең маңыздызы - адамдардың жүрегінен орын табу.
            Біздің компанияда тәжірибесі мол, өз ісінің шебер мамандары
            қызмет жасайды. Біздің сфера бүкіл Қазақстанда даму үстінде!</p>
        </div>
        <div class="col-md-3 footer-grid">
          <h4 class="footer-head">Көмек</h4>
          <ul>
            <li><a href="howitworks.html">Қалай жұмыс істейді</a></li>
            <li><a href="faq.html">Сұрақтар</a></li>
            <li><a href="feedback.html">кері байланыс</a></li>
            <li><a href="contact.html">Контакт</a></li>
            <li><a href="typography.html">Shortcodes</a></li>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>
```

Б қосымшасының жалғасы

```
<div class="col-md-3 footer-grid">
  <h4 class="footer-head">Ақпарат</h4>
  <ul>
    <li><a href="regions.html">Мекен-жай</a></li>
    <li><a href="terms.html">Қолдану ережелері</a></li>
    <li><a href="popular-search.html">Танымал іздеулер</a></li>
    <li><a href="privacy.html">Құпиялылық саясаты</a></li>
  </ul>
</div>
<div class="col-md-3 footer-grid">
  <h4 class="footer-head">Байланысу</h4>
  <span class="hq">Біздің штабымыз</span>
  <address>
    <ul class="location">
      <li><span class="glyphicon glyphicon-map-marker"></span></li>
      <li>Алматы қаласы, Мауленова 85</li>
      <div class="clearfix"></div>
    </ul>
    <ul class="location">
      <li><span class="glyphicon glyphicon-earphone"></span></li>
      <li>+7 707 417 35 17</li>
      <div class="clearfix"></div>
    </ul>
    <ul class="location">
      <li><span class="glyphicon glyphicon-envelope"></span></li>
      <li><a
href="mailto:info@example.com">syreken.ss@gmail.com</a></li>
      <div class="clearfix"></div>
    </ul>
  </address>
</div>
<div class="clearfix"></div>
</div>
<div class="footer-bottom text-center">
  <div class="container">
    <div class="footer-logo">
      <a href="index.html"><span>Kaz</span>Market</a>
    </div>
    <div class="footer-social-icons">
      <ul>
        <li><a class="facebook" href="#"><span>Facebook</span></a></li>
```

Б қосымшасының жалғасы

```
<li><a class="twitter" href="#"><span>Twitter</span></a></li>
    <li><a class="flickr" href="#"><span>Flickr</span></a></li>
    <li><a class="googleplus" href="#"><span>Google+</span></a></li>
    <li><a class="dribbble" href="#"><span>Dribbble</span></a></li>
</ul>
</div>
<div class="copyrights">
    <p>KazMarket 2019</p>
</div>
<div class="clearfix"></div>
</div>
</div>
</footer>
<!--footer section end-->
</body>
</html>
```

[illegible]

Ғылыми жетекшінің пікірі

Дипломдық жоба

Сейдазимов Сырым Байболұлы

5B060200 – Информатика

Тақырыбы: Ұялы телефондар мен компьютерлерге арналған маркеттің клиенттік бөлігін құру.

Диплом жобасын жасаушы Сейдазимов Сырым Байболұлы ұялы телефондар мен компьютерлерді сатушы мен іздеуші адамдарды байланыстыруды оңтайландыруға мүмкіндік беретін веб-сайт жұмыс жасауына қажетті, пайдаланушыларға түсінікті, оңтайлы интерфейс құру және хабарландыруды енгізу, қарау және пікір қалдыру міндеттері қойылған. Веб-сайт құру барысында Atom бағдарламалау ортасы және JavaScript бағдарламалау тілі мен оның кітапханасы React қолданылуы тиіс болатын.

Аталған технологиялар веб-сайтты құруда джуниор бағдарламалаушыларға таптырмас құрал және дүниежүзінде алдыңғы орында. Сондай-ақ клиенттен келген сұрауларды серверге жіберу мүмкіндіктеріне ие, сондықтан көптеген ірі компанияларда кеңінен қолданылады.

Сейдазимов Сырым Байболұлы веб-сайтты, бұл сайтқа арналған Atom бағдарламалау ортасы, JavaScript бағдарламалау тілі технологияларын жақсы меңгере отырып веб-сайтты құру жолдарын жүзеге асырды. Бағдарлама құру туралы теориялық ақпараттар келтірілген. Нәтижені әрі қарай өңдеу және жақсарту талап етіледі.

Дипломдық жобаның негізгі мазмұнын құрайтын барлық нәтижелер студенттің өз бетімен алынған, ал жетекшіге мәселенің қойылуы, кеңестер мен нәтижелерді талдау ғана тиісті.

Қорытындылай келе, дипломдық жоба 5B060200 – «Информатика» мамандығының бітіру жұмыстарына қойылатын талаптарына сәйкес және 5B060200 – «Информатика» мамандығы бойынша «Жаратылыстану бакалавры» академиялық дәрежесін тағайындауға болады және дипломдық жобаны қорғауға жіберіледі деп есептеймін.

Ғылыми жетекші

«Программалық инженерия»

кафедрасының оқытушысы, лектор

 Д.А. Баймбетов

«17» 05 2019 ж.