

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

**ҚОРҒАУҒА ЖІБЕРІЛДІ**

Кафедра меңгерушісі

Техника ғылымдарының кандидаты,  
ассистент-профессор

 Р. Юнусов

«21» мае 2019 ж.

Дипломдық жобаға  
**ТҮСІНІКТЕМЕЛІК ЖАЗБА**

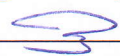
Тақырыбы: “ Оқытушыларға оқушылардың білім сапасын анықтауға арналған  
сайттын құру ”

5B060200 – «Информатика» мамандығы

Орындаған

А.Р. Мейрамбек

Ғылыми жетекші,  
сениор-лектор

 А.Д. Куникеев  
«13» 05 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

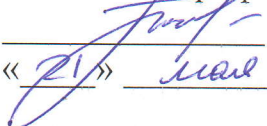
Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

5B060200 – «Информатика»

**БЕКІТЕМІН**

ПИ кафедра меңгерушісі,  
техника ғылымдарының кандидаты,  
ассистент-профессор

 Р. Юнусов  
«21» мамыр 2019 ж.

**Дипломдық жоба орындауға  
ТАПСЫРМА**

Білім алушы Мейрамбек Арман Ролланұлы

Тақырыбы Оқытушыларға оқушылардың білім сапасын анықтауға арналған сайттың құру

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

Аяқталған жобаны тапсыру мерзімі 2019 жылғы «21» мамыр

Дипломдық жобаның бастапқы берілістері Ұсынылатын дипломдық жобада Оқытушыларға оқушылардың білім сапасын анықтауға арналған сайттың құру. Веб-сайтты құру барысында серверлік бөлімде PHP бағдарламалау тілін және MySQL деректер қорын, әзірлеу бөлімінде CSS тілінің Bootstrap 4 тірегін қолдандым.

Дипломдық жобада қарастырылатын мәселелер тізімі:

а) Жалпы бөлім;

ә) Негізгі бөлім;

б) Жобалау бөлім;

в) Веб бетті құру және оны іске асыру.

Сызба материалдарының тізімі (міндетті сызбалар дәл көрсетілуі тиіс)

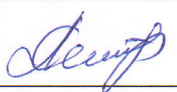

Жобаның презентациялық 20 слайды ұсынылған.

Ұсынылатын негізгі әдебиет 19 әдебиеттер тізімінен.

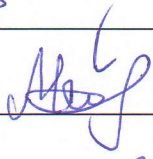
Дипломдық жобаны дайындау  
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекші мен кеңесшілерге көрсету мерзімдері	Ескерту
Жалпы бөлім	18.02.2019	жоқ
Қолданылған бағдарламалық қамтама	10.03.2019	жоқ
Жобалау бөлімі	28.04.2019	жоқ
Дипломдық жұмыстың түсініктемелік	12.05.2019	жоқ

Дипломдық жоба бөлімдерінің кеңесшілері мен норма бақылаушының аяқталған жобаға қойған қолтаңбалары

Бөлімдер атауы	Кеңесшілер аты, әкесінің аты, тегі (ғылыми дәрежесі, атағы)	Қол қойылған күн	Қолы
Норма бақылау	Алғожаева Р. Сениор - лектор	17.05.2019	
Бағдарламалық бөлімі	Сман Н.О. Ассистент	17.05.2019	

Ғылыми жетекші \_\_\_\_\_  А.Д. Куникеев

Тапсырманы орындауға алған білім алушы \_\_\_\_\_  А.Р. Мейрамбек

Күні \_\_\_\_\_ «13» 05 2019 ж.





**Қ.И. Сәтбаев атындағы Қазақ ұлттық техникалық зерттеу университеті**

**Мамандығы 5B060200 – Информатика**

**Студент Мейрамбек Арман**

**Тақырыбы** Оқытушыларға оқушылардың білім сапасын анықтауға арналған сайт

### **ҒЫЛЫМИ ЖЕТЕКШІНІҢ СЫН-ПІКІРІ**

Диплом жобасын жасаушы Мейрамбек Арманның алдына оқытушыларға оқушылардың білім сапасын анықтауға арналған сайтты өңдеу тапсырмасы қойылған.

Дипломдық жобада оқытушыларға оқушылардың білім сапасын анықтап, күнделікті білім деңгейін анықтауға мүмкіндік беретін веб-сайты құрылған.

Жұмыс барысындағы басты мәселелер: репетиторлар сайттарын зерттеп талдау, веб-сайт құру үшін қажетті тілдер мен ыңғайлы әдісті тандау, диаграммалар сызу, деректер қорын құру, сайт дизайнын ойластыру. сайтқа қажетті технологияларды орнату және кітапханаларды орнату, сайтқа оңтайландырылған әдісті таңдау болып табылады. Бұл қазіргі заманауи ең өзекті және келешегі бар технологияларды қолданудағы маңызды мәселелердің бірі.

Қорытындылай келе, осы зерттеу жұмысын орындау барысында Мейрамбек Арман Ролланұлы жетекшінің көмегімен жобасын ойдағыдай атқарып шықты.

Менің пікірімше, диплом жазушы алдына қойылған тапсырманы толығымен орындады және ақпараттық жүйелердің заманауи технологияларын меңгергендігін көрсетті.

Жоба жетекшісі ретінде бұл дипломдық жобаны өз деңгейіне сәйкес деп есептей отырып Мейрамбек Арманға 5B060200 – «Информатика» мамандығы бойынша «Жаратылыстану бакалавры» академиялық дәрежесін тағайындауға болады деп есептеймін.

**ҒЫЛЫМИ ЖЕТЕКШІ**

**«Программалық инженерия»**

**кафедрасының сениор-лекторы «13» 05 2019 жыл**



**А.Д. Куникеев**

## АҢДАТПА

Бұл дипломдық жобада оқытушыларға оқушылардың білім сапасын анықтауға арналған сайт қарастырылады. Ақпараттық жүйе толықтай CSS стильдер тілінің негізі, Bootstrap – құралдар жиынтығы, Grid жүйесі, PHP скрипті бағдарламалау тілі, PhpMyAdmin веб-қосымшасы, Apache HTTP сервері, Open Server, MySQL – деректер қоры, GitHub репозиторий, JavaScript программалау тілі, jQuery, Ajax технологияларын қолдану арқылы жасалды.

Дипломдық жоба кіріспеден, үш бөлімнен, 29-суреттен, қорытындыдан, пайдаланылған әдебиеттер тізімінен және қосымшадан тұрады. Кіріспеде жобаның мақсаты, өзектілігі мен міндеттері көрсетіледі.

Бірінші бөлімде қазіргі таңдағы репетиторлардың қажеттілігі, репетиторлар туралы сайттарды зерттеу және салыстыру түсіндірілген.

Екінші бөлімде қолданылатын ақпараттар, жобаның логикалық және бағдарламалық құрылымы туралы сипаттамалар берілген.

Үшінші бөлімде диаграммалар мен программаның құрылымы туралы сипаттамалар берілген.

Құрастырылған ақпараттық жүйені жасау мен ендіру жолдары сәйкестендіріліп, жобада анық айтылған. Жобаны іске асыру барысында қойылған мақсатқа қол жеткізіліп, сәйкес мәселелері шешілді – «Оқытушыларға оқушылардың білім сапасын анықтауға арналған сайт» жүйесі құрастырылды.

Бұл мақалада PhpStorm бағдарламалау ортасында PHP бағдарламалау тілінде және көптеген қосымшалардың көмегімен әзірленген бағдарлама.

## АННОТАЦИЯ

Этот дипломный проект предоставляет сайт для репетиторов с возможностью узнать оценки школьников. Информационная система полностью разработана с использованием на основе языке стилей CSS, наборах инструментов Bootstrap, Grid-система, языке программирования PHP-скриптов, веб-приложении PhpMyAdmin, HTTP-сервере Apache, Open Server, базе данных MySQL, репозитории GitHub, языке программирования JavaScript, jQuery, технологии Ajax сделано через.

Дипломный проект состоит из введения, трех разделов, рис.29, заключения, списка использованной литературы и приложения. Во введении указывается цель, актуальность и задачи проекта.

Первая часть объясняет необходимость современных репетиторов, обзор и сравнение учебников.

Второй раздел содержит информацию о логике и программной структуре проекта.

Третий раздел содержит описания диаграмм и структуры программного обеспечения.

Способы создания и реализации скомпилированной информационной системы определены и четко указаны в проекте. В ходе реализации проекта были достигнуты цели и решены соответствующие проблемы - разработана система «Сайт для репетиторов с возможностью узнать оценки школьников».

Эта статья представляет собой среду программирования PhpStorm на языке программирования PHP и разработана для многих приложений.



## ANNOTATION

This diploma project provides website for identifying students' quality of education. The information system is based entirely on CSS style language basics, Bootstrap toolkits, Grid system, PHP scripting programming language, PhpMyAdmin Web application, Apache HTTP server, Open server, MySQL database, GitHub repository, JavaScript programming language, jQuery, Ajax technology made through.

The diploma project consists of an introduction, three chapters, 29th form, conclusion, list of used literature and annex. In the introduction, the purpose, relevance and objectives of the project are indicated.

The first part explains the need for modern tutors, review and comparisons of tutorials.

The second section provides information about the logic and program structure of the project.

The third section provides descriptions of charts and software structure.

Ways to create and implement a compiled information system are identified and explicitly stated in the project. During the project implementation the goals were achieved and the corresponding problems were solved - a system of " Website for identifying students' quality of education" was developed.

This article is a PhpStorm programming environment in the programming language of PHP and is developed with many applications.



## МАЗМҰНЫ

	Кіріспе	8
1	Жалпы бөлім	9
1.1	Қазіргі таңдағы репетиторлардың қажеттілігі	9
1.2	Репетиторлар туралы сайттарды зерттеу және салыстыру	11
2	Негізгі бөлім	13
2.1	CSS стильдер тілінің негізі	13
2.1.1	Bootstrap – құралдар жиынтығы	13
2.1.2	Grid жүйесі	14
2.2	JavaScript программалау тілі	16
2.2.1	jQuery	18
2.2.2	Ajax	19
2.3	PHP скрипті бағдарламалау тілі	20
2.4	PhpMyAdmin веб-қосымшасы	23
2.5	Apache HTTP сервері	23
2.6	Open Server	24
2.7	MySQL – деректер қоры	25
2.8	GitHub репозиторий	28
3	Жобалау бөлімі	30
3.1	Rational Rose туралы түсінік	30
3.1.1	Қолдану диаграммасы	32
3.1.2	Тізбектер диаграммасы	34
3.1.3	Класстар диаграммасы	37
3.1.4	Қызмет диаграммасы	38
3.2	Программа құрылымы	39
	Қорытынды	45
	Қолданылған әдебиеттер тізімі	46
	А Қосымшасы	47
	В Қосымшасы	49
	Спецификация	55

## КІРІСПЕ

Білімге инвестиция салу – ең табысты инвестициялардың бірі, ал репетитор бала мен ғылым әлемі арасындағы делдал болып табылады. Репетиторлық қызметтер математика, физика, химия және одан да күрделі пәндерді жетілдіру, жұмысқа тұру үшін немесе ағылшын және басқа шет тілдерін үйрену үшін қажет.

Репетиторлар мен студенттер арасында екі жақты байланыс жасау идеясы бізге шетелден келді. Бұл идея өткен ғасырда Еуропа мен Америкада дами бастады, және мұғалімдерге арналған сайттардағы табыстар осындай сайттардың қажет екенін көрсетеді.

Бүгінгі күні газет хабарландырулары бұрынғыдай танымал емес. Біз виртуалды әлемнен ажырамаймыз, сондықтан «репетитор іздестіреміз» деп хабарландыруға тапсырыс беру арқылы репетитор тарту екіталай. Бізге мұғалімді тауып, тандап алатын ақылды іздеу қозғалтқышына жүгіну әлдеқайда ыңғайлы. Осындай жүйені танымал ету міндетін репетиторлық сайттар қабылдады. Олардың әрқайсысында мұғалімдердің базасы бар, ол мұғалім туралы ақпарат береді және кері байланыс жылдамдығын қамтамасыз етеді.

Жалпы база – ең танымал репетиторлар алдыңғы қатарда болатын каталог. Әрбір сайт өзінің тәлімгерлік тәжірибесін, оның жетістіктерін, марапаттарын және т.б. ескере отырып, өзінің рейтингілік жүйесін қалыптастырады. Үйден шықпай-ақ мұғалімнің қысқаша мазмұнын оқи аласыз.

Оқытушыларға арналған сайттар клиенттер үшін мұғалімдерді іздестіреді, сондай-ақ жаңадан келген репетиторлар мен клиенттерді тіркейді. Репетиторды таңдау әлеуметтік желілердің классикалық қағидасына негізделген – бір түймені басу арқылы табу. Одан кейін болашақ клиент пен мұғалім көрсететін қызметі және қажетті мәліметтерді түсіндіреді. Екі жақ сабақтардың қайда өткізілетінін шешеді: мұғалімнің үйінде немесе баланың үйінде, соңғы кезең-кездесу және одан әрі серіктестік.

Осылайша, репетиторлық сайттар қызметінің мақсаты барлық емтихандарда, тесттерде және өзіндік жұмыста жоғары балл алуға көмектесетін мұғалімді ыңғайлы және тез табу болып табылады. Тәжірибелі маман оқу үрдісін жеделдетіп, ынталандырады және табысқа жету үшін нақты бағдарламаны әзірлейді.

Ал менің дипломдық жұмысымның мақсаты осындай репетиторларға арналған сайт жасау болып табылады.

Берілген дипломдық жұмыс кіріспеден, 3 бөлімнен, қорытындыдан, қолданылған әдебиеттер тізімінен және қосымшалардан тұрады.

## 1 Жалпы бөлім

### 1.1 Қазіргі таңдағы репетиторлардың қажеттілігі

Халықаралық практикаға сәйкес «жеке репетиторлық» термині «негізгі білім беруден басқа оқушы өз қалтасынан төлеп мектеп пәндерін оқыту» дегенді білдіреді. Бүгінгі күні ұлттық білім беру сапасының төмендеуі кесірінен қоғам алдында репетитор – ақылды, мейірімді, оқуға қатысты барлық сұрақтардың шешімі болып табылады.

Білім беру саласындағы зерттеушілер Қазақстандағы жеке репетиторлық сабақтың маңызды кезеңнен өткенін және білім беру жүйесіндегі дағдарыстық құбылыстар туралы айтады. Егер жеке сектордағы қызмет халықтың 25%-нан астамын пайдаланады. Бұған себеп көлеңкелі нарық, білімге тең емес қолжетімділік, оқыту сапасының төмендігі.

Бірнеше жыл бұрын қазақстандық сегментті Ашық Қоғам Институтының Білім беруді Қолдау Бағдарламасының (Будапешт) өкілдері зерттеді. Сауалнамаға педагогтар мен бірінші курс студенттері қатысты.

Сауалнаманың нәтижесінде Жоғары оқу орнына түсетін әрбір екінші студент репетиторлық қызметтерді пайдаланған екен. Сонымен қатар, мұғалімдердің қызметтерін пайдаланғандардың 68%-ы соңғы оқу жылында жүйелі түрде оқып жүрген, 70%-ы аптасына екі-төрт сағатқа, бір немесе екі пәннен, 28%-ы үш немесе одан да көп пәннен айналысқан, талапкерлердің 62%-ы дайындық курстарына қатысып, мұғаліммен қатар айналысқан.

Сол зерттеуге сәйкес, Қазақстанда математика пәнінің мұғалімдері ең көп сұранысқа ие (67%), тарих және физика екінші орынды алады (36%), қазақ тілі мен әдебиеті үшінші орынды иеленеді (18%).

Оқу сабақтарының негізгі мақсаты:

- қабылдау емтихандарына дайындық;
- екінші орында – тақырыпты жақсы түсіну;
- үшінші деңгейдегі білімдерін толтыру.

Біздің еліміздегі репетиторлықтың пайда болуының себебі қазақстандық мұғалімдер тұрғысынан айтқанда – орта білімнің төмен деңгейі. Бұл себепті респонденттердің 57%-ы айтады. Мұғалімдердің 37,8%-ының пайымдауынша екінші маңызды себебі мемлекеттің жоғары білім саласындағы қолдауының әлсіздігі. Бұл университеттердің тегін бөлімдеріндегі шектеулі орындарда және тиісінше, олар үшін бәсекелестіктің артуымен көрінеді. Респонденттердің 21,6%-ы репетитордың пайда болуының маңызды себептерінің бірі репетиторлықпен айналысатын мұғалімдердің қосымша табыстар табуы қалауы болып табылатынын мәлімдеді.

Қазақстанда репетиторлық қызметтердің құны сағатына 1000 теңгеден 5000-ға дейін жетеді, бұл оқушы мектептегі білім деңгейін толтырады ма, немесе университетке немесе ҰБТ-ға түсуге байланысты. «Баға ұстазы» сұранысы бойынша Интернеттегі іздеу жүйесі алдағы үш беттен тұрады. Мұнда

тек жеке ұстаздар ғана емес, сонымен қатар үлкен ұстаздар базасы бар оқу орталықтары да бар. Мәселен, ата-анасы дайындаған бірінші сыныпқа оқуға шешім қабылдаған мектепке дейінгі жастағы балалар үшін, мәселе бағасы сағатына 1000 теңге. Мектеп оқушылары үшін пәнге қарамастан 1000 теңгеден 2000-ға дейін. Яғни, әртүрлі тәрбиешілер әртүрлі бағаға ие. Бір мұғалім 2 мың теңге тұратын математика сағатына, ал қалған жартысы аз. Тілдермен қатар. 45 минутта табылған ең жоғарғы баға 3500 теңге (ағылшын) болды.

Университеттің кіруіне келер болсақ, мұндағы баға 2000-нан 5000 теңгеге дейін болады. Мұның барлығы үміткер таңдаған мамандыққа байланысты және мұғалімнің кәсібилігі маңызды рөл атқарады.

Алайда, жеке сабақтар бар, онда сағатына 7000 теңге болады. Құны үшін тілек білдірушілер ағылшын тілінде математиканы үйретеді.

Тұтастай алғанда, егер алдыңғы жылдағы репетиторлық қызметтердің құнын салыстырсақ, баға бірден өзгерген жоқ. Тәрбиешілер нарығы жыл сайын кеңейіп келеді және бәсекелестік Қазақстан халқына арналған. Дегенмен, жақсы мұғалімдер, әрқашан да, бағаланады, сондықтан айтатын болсақ, қолынан ұстап тұру керек. Мұнда белгілі бір тәрбиешіден білім алғысы келетіндердің кезегі де бар.

«Бір жыл бойы Вадим Валентиновичтің сабағына бара алмадық. Ақыр аяғында біз бұл ұсынысты орындадық», – дейді Антонина Владимировна форумдардың бірінде – Влад ҰБТ-ны тапсыруға дайындалды. Вадим Валентинович кез-келген күрделі міндеттерді оңай және тез шешуді түсіндіре алды. Кез-келген уақытта, күндіз-түні ұлым мұғалімге жаза алды және ол бірден жауап берді! Влад мектепте өз білімін көрсеткен кезде математика мұғалімдері таңқалды. Ол ҰБТ-ны керемет түрде тапсырып, грантқа университетке түсті. Ол жоғары математиканы шемішкідей шағатын болды».

«Түсіну керек, сабаққа бір сағат уақытыңды жұмсап және ешкімнің алдында жауап бермеу әлдеқайда оңай, – дейді ағылшын тілі мұғалімі Марина. Балаларды іріктеуде ешқандай қиындық жоқ, бірақ менде сабақтардың сағаты 2 750 теңгені құрайды. Мұнда ауызша реклама жұмыс істейді, мен бұл жарнаманы көбірек ұнатамын. Мен ИП ашуды, мемлекетке төлеуді қаламаймын. Салық пен зейнетақы мұғалімнің лауазымдық жалақысынан менің жалақымнан алынады. Кейінірек зейнеткерлік ақшамды көретініме деп күмәнданамын».

«Репетиторлар өз қызметтерінің бағасын сирек көтереді. Бүгінгі таңда баға ауқымы өте үлкен. Бірақ алтын әріп бар. Сағатына 2000 теңгеге кез-келген мектеп тақырыбын таба аласыз. Әрине, белгілі бір тәрбиешіге қандай оқыту сапасын беретінін білу әрқашан қажет», – дейді орыс тілі мен әдебиетінің 20 жылдық тәжірибесі бар оқытушы Анна Курина. Қазір көптеген жас адамдар бар, диплом иегері және үстірт білімі бар адамдар көп емес, олар көп табуға болады деп ойлайды. Бірақ, өкінішке орай, іргелі білімі аз мамандар бар. Бір жағынан бүгінгі күні көптеген адамдар тьюторларға бет бұрғанына қуаныштымын. Адамдар таңдауға және таңдауды қалайды. Бірақ, екінші жағынан, білім беру жүйесі үшін қорқамын».



Кейбір мұғалімдер жекеменшік сабақтарда көп табыс табады деп есептейді, өзін-өзі жұмыспен қамтығандар қатарына қосып, мектептерді тастап кетеді.

«Мен жарты күн, аптасына 5 күн жұмыс істеймін және 200 мың теңгеден астам табыс табамын. Мен жұмысқа күніне 3-4 сағат жұмсамаймын, мен ешкімге тәуелдімін», – дейді физика пәнінің мұғалімі Валентин Акунин (аты-жөні мұғалімнің өтініші бойынша өзгертіледі). Сонымен қатар студент үйренгісі келеді, мен оның білімінің жетістігін көремін. Мұғалім үшін – бұл ең бастысы! Сондықтан, мен өзімнің сабақта тақырыпты түсіндіргеннен гөрі жеке репетиторлық қызметті артық көремін, онда адамдардың көпшілігі өз көздеріне бей-жай қарамайды. Ай айының аяғында жалақыны көріп, не жылар не күлерімді білмеймін».

## **1.2 Репетиторлар туралы сайттарды зерттеу және салыстыру**

Онлайн-репетитор сайты – бүгінгі таңда жасөспірімдер арасында ең қажетті жұмыс орындарының бірі. Бұл сіздерге жақсы жұмысқа қабілеттілік пен икемділікті береді, олардың көпшілігі тіпті тұрақты жұмыс ретінде қабылдайды. Көптеген сайттар жақсы табыс көзін табуға көмектесе алады және олардың әрқайсысы өзінің артықшылықтары мен кемшіліктері бар.

Біз ғаламтордағы мұғалімдер мен студенттерге арналған тамаша веб-ресурстарға толы екенін білеміз. Алайда, интернетте өмір сүрмейтін болсаңыз, ең жақсы сайттарды іздеу қиынға соғуы мүмкін. Бұл «ондықтың» тізімі емес – бұл мен жүйелі түрде пайдаланатын немесе қызықтыратын сайттардың тізімі. Сізге сәйкес келетін келесі жұмыстар тізімінде, жаратылыстанудағы білім беру ресурстарынан және шынайы өрістердегі журналдардан сандар туралы ақылсыз бейнелерге кез-келген нәрсе таба аласыз! Таңдаулы ғылыми веб-ресурстарыңызды түсініктемелер бөлімінде бөлісіңіз! Қазіргі таңда көпшілік білім алу дәрежесіне жағдайлар жасалған. Бірақ веб-сайттардың көпшілік көзі тек қана тіркеліп, онлайн түрде оқытылмайды. Мысалы, білім алушы кірген жағдайда, дерек көздерін алып оқып соған байланысты тесттарды тапсыру арқылы өз білім сапасын тексеру дәрежесінде көрсетілмеген. Көпшілік сайттар онлайн түрде көрсетіледі. Яғни, білім беруші тіркеліп, білім беретін бағытын жазып кетеді. Соған байланысты білім алушы сол керекті пәндері арқылы білім берушілерді іздеп, олардың мекен-жайы бойынша немесе білім беруші оның мекен-жайы бойынша білім береді. Бірақ дегенмен қашықтықтан оқыту саласынан бізде веб-сайттар дамымаған, желіде немесе видео курстарда болмаса. Онлайн түрде білім алушының дәрежесің тексеріп отыратын сайттар қажет.

Қазіргі таңдағы актуалды репетиторлар сайттарын зерттей келе олардың басым бөлігінің құрылымы мен жұмыс жасау бағыты бірдей. Яғни, білім беруші(репетитор) сайтқа тіркеліп өзі туралы ақпарат қалдырады, ал оқушылар

болса сол сайтқа кіріп өзі іздеген бағыттағы мұғалімді тандайды. Содан соң мұғаліммен байланысқа шығу арқылы білім берушінің мекен-жайы бойынша немесе білім алушының мекен-жайы бойынша дайындық орнын белгілейді.

Ал Education сайтындағы мүмкіндіктер басқаша болмақ:

- сайтты реттеп білім беруші мен білім алушыны тіркейтін администратор;
- білім беруші администратормен байланысқа шығу арқылы тіркеліп өзі туралы ақпаратты жібереді;
- білім алушы администратор арқылы тіркеледі;
- администратор, білім беруші және білім алушы өз профильдерін және құпиясөздерін өзгерте алады.
- әр білім беруші сабағына қатысты материалдарды жүктей алады;
- білім беруші сайтқа тест жүктеу арқылы өзінің оқушыларының білімін тексеріп отыра алады;
- білім алушы өзіне керекті материалдарды жүктеп дайындала алады;
- білім алушылардың тесттегі бағаларына байланысты білім берушілердің рейтингі құрылады.

Бұл проектіні құрудың себебі, қазіргі таңда белгілі бір кеңсенің дайындығы бойынша және ақпарат беру және жіберу жағы аздап қиындау деп айтсақ та болады. Сол себепті өткен сабағы бойынша және сол өткен сабақтарын білім алушы қайталап отыра алатындай жағдай жасалынды. Ал білім берушіге ақпаратты алмасу өте қолайлы деңгейге түсті. Мұғалім тапсырманы жіберер кезінде ыңғайлы түрде қарастырып және апта сайын білім алушыларының нәтижесін тексеріп отыра алады. Соған орай білім алушылардың алған бағаларына сай репетиторлардың рейтингі шығып отырады. Олар әрине апта сайын өзгеріп отырады. Администратор жаңа мұғалімдер мен оқушыларды тіркеп отырып, оларға жеке кабинеттеріне кіруіне рұқсат береді. Ал мұғалім мен оқушылар сол өзінің кіру бетіне «логин» мен «құпиясөзін» администратордан алған соң, жеке кабинетіне кіруге мүмкіндік алады. Сол арқылы олар өз мұғалімінен алған білімдері бойынша ақпаратты көріп, қайталап апта сайын тест тапсыру мүмдігіне ие болады. Осы себептен олар білім сапасын жоғары деңгейге көтере алады. Ал оқытушыға ыңғайлы болатын себебі ол ақпараттарды сонда күнделікті еңгізу арқылы және тестті салу арқылы өзінің оқушыларының білім сапасын тексеріп отыра алады. Бұл веб-сайттың қазіргі таңдағы қолданысы қолайлы. Себебі жеке кеңсе орындардың дамуына және танымалдығына себепкер болады. Оқушылардың білім алу жағы қазіргі таңдағы қабылдаулары өте төмен деп айтуға болады. Себебі олар желінің дамуына байланысты видео сабақтар немесе сілтемелер арқылы тез қабылдайды. Сол себепті олар өткен сабақтары бойынша кеңсе орында өтіп, оны қайтадан қайталап, оқып дайындалуына мүмкіндік береді.

## 2 Негізгі бөлім

### 2.1 CSS стильдер тілінің негізі

CSS – Стиль кестелерін (Cascading Style Sheets) каскадтау. CSS – шын мәнінде HTML құжаттарының көрсетілуін анықтайтын стиль тілі. CSS – шрифттермен, шекарамен, сызықтармен, биіктіктерімен және дисплей элементтерінің енімен, өңдік суреттермен, элементтерді орналастырумен және басқа да көптеген нәрселермен жұмыс істейді.

Егер HTML беттің мазмұнын құрастыру үшін қажет болса, онда бұл құрылымды мазмұнды пішімдеу үшін CSS қажет.

CSS-ні пайдалану сайтымыздағы барлық беттер бірдей стильде сақталатынына сенімді болу үшін белгілі бір css-файлдарында жеке торап беттеріне мәнерлер қоюға мүмкіндік беретін жоғары сапалы веб-сайттарды жасауды жеңілдетеді.

#### 2.1.1 Bootstrap – құралдар жиынтығы

Bootstrap – бастапқыда Twitter-тің ішкі қолданысы үшін жасалынған «Twitter Blueprint» атты жұмыс жасау үшін жасалған CSS пішімі, бірақ кейіннен қоғамдық қолжетімділік үшін жарияланып, «Bootstrap» деп аталатын кеңейтілген, даму үшін жақсы құрал болды.

Bootstrap мүмкіндіктері:

- сайт бетінің орналасуының жоғары жылдамдықты дамуы. Bootstrap дайын шешімдер мен элементтердің үлкен жиынтығын қамтиды;
- кросс-браузердің үйлесімділігі және торапқа бейімделуі. Барлық құрылымдық элементтер барлық құрылғыларға бейімделеді және барлық қазіргі заманғы браузерлерде дұрыс көрсетіледі;
- пайдалануға ыңғайлы. Тіпті HTML және CSS-ң базалық білімі бар адам, веб-беттерді еркін түрде жасай алады;
- оқудың қарапайымдылығы. Жүктеу бумасы дайын коды көптеген мысалдармен өте жақсы құжаттамаға ие.

Bootstrap-ты WEB-тің негізі деп айтуға болады, себебі ол құрамында CSS, HTML және JavaScript компоненттері бар, ал үшінші нұсқасында өзіндік икондары бар. Bootstrap-та 250-ден астам иконкасы бар.

Bootstrap негізгі құралдары:

- торлар – мысалы, 140 px баған ені дереу қолдануға болатын алдын ала анықталған баған өлшемдері, құжаттың CSS сипаттамасында қолданылуы мүмкін .span2 (.col-md-2 құрылымның үшінші нұсқасында);
- үлгілер – тіркелген немесе резеңке құжат үлгісі;

- типография – шрифт сипаттамасы, қаріптер үшін белгілі бір сыныпты анықтау, мысалы, код, тырнақша және т.б;
- медиа – кейбір бейнелер мен бейнелерді басқаруды қамтамасыз етеді;
- кестелер – кесте құрастыру құралы, сұрыптау функциясын қосу үшін;
- пішіндер – олармен кездесетін нысандардың және кейбір оқиғалардың дизайны бойынша сабақтар;
- навигация – панельдер, қойындылар, беттер арқылы шарлау, мәзірлер мен құралдар тақталары үшін дизайн сыныптары;
- ескертулер – диалогтық терезелерді, шақыруларды және қалқымалы терезелерді жобалау.

Bootstrap CSS және HTML саласындағы қазіргі заманғы тәжірибелерді пайдаланады, сондықтан ескі браузерлердің қолдауымен абай болу керек.

### 2.1.2 Grid жүйесі

Тор – көлденең және тік сызықтардың көлденең жиынтығы деп аталады. Оның бірі бағандарды, ал екіншісі жолды анықтайды. Элементтерді тиісінше торға, жолдарға және бағандарға орналастыруға болады.

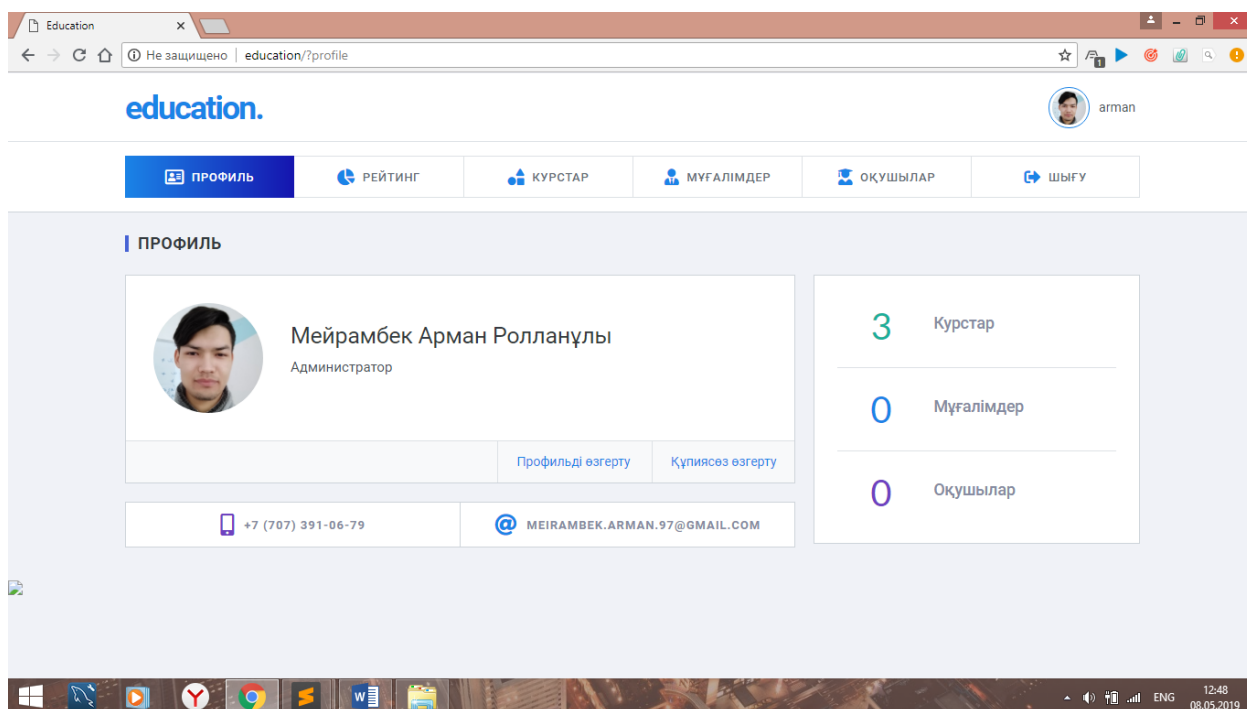
Тордың келесі функциялары бар:

- бекітілген және икемді диапазон өлшемдері. Тіркелген жолақ өлшемдері бар торды жасай аласыз – мысалы, пикселдерді пайдалану. Сондай-ақ, пайыздық мөлшерді немесе өлшемді жаңа өлшем бірлігін пайдалана отырып, икемді өлшемдері бар тор құра аласыз – бұл мақсат үшін арналған;
- бөлімнің орналасқан жері. Элементтерді белгілі бір тораптағы орынға сызық нөмірлері, аттар немесе Тор аймағына байланыстыру арқылы орналастыруға болады. Тор жүйесінде торда анық позиция жоқ элементтерді орналастыруды басқарудың алгоритмі де бар;
- мазмұнды сақтау үшін қосымша жолақтар жасау. Тор торабымен айқын торды анықтай аласыз, бірақ ерекшелігі, қажет болғанда қосымша жолдар мен бағандарды қосып, жарияланған тордан тыс қосылған мазмұнға қолданылады. Құрамында «көп контейнерлерде орналастырылатын бағандар» сияқты элементтер қосылады;
- тегістеуді басқару. Тор жүйесінде элементтердің тор торабына орналастырылғаннан кейін қалай реттелетінін және бүкіл торды теңестіруін бақылау үшін, олардың орналасу функциялары бар;
- айырбастауға арналған мазмұнды басқару. Бірнеше элемент тор торына орналастырылуы мүмкін, немесе аумақтар бір-бірін жабуы мүмкін. Бұл пакет z-индексі арқылы бақылануы мүмкін.

Тор – қуатты спецификация және CSS-дің басқа бөліктерімен бірге, мысалы flexbox, бұрын CSS-де жасалмаған орналасуларды жасауға көмектеседі.

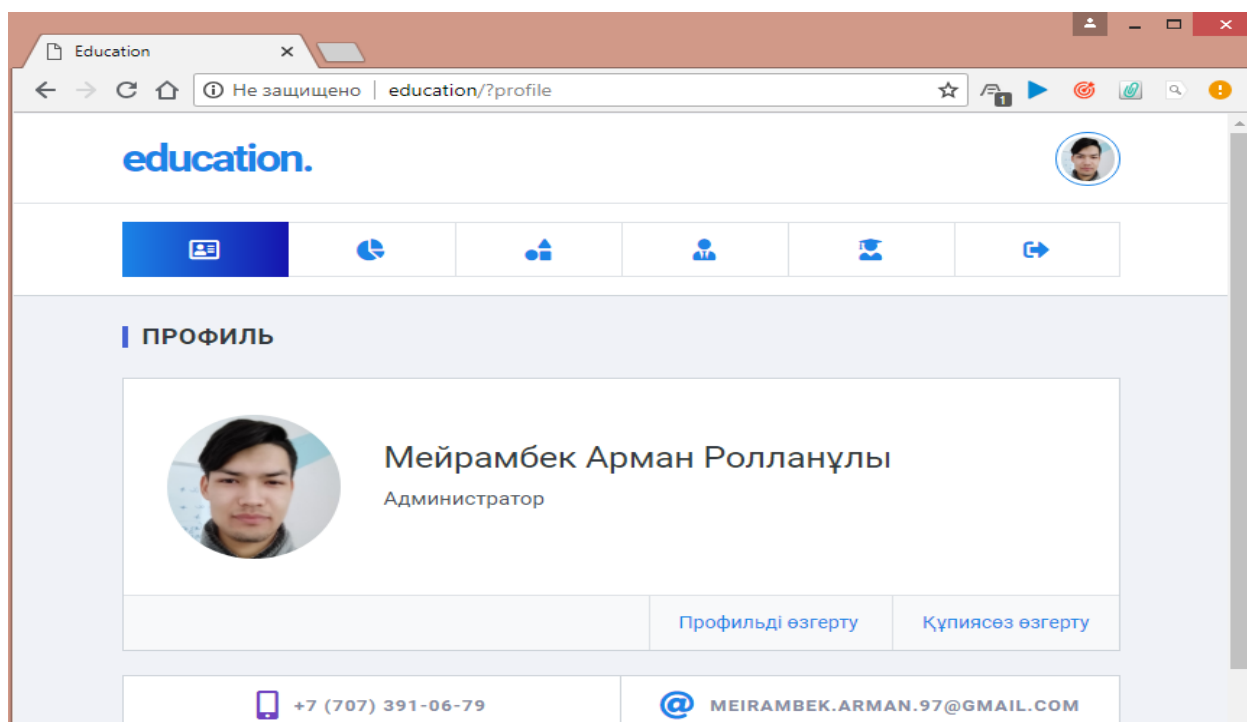


Барлығы тор торындағы торды жасаудан басталады. Grid жүйесінің браузердегі көрсетілімі 1.1-суретінде көрсетілген.



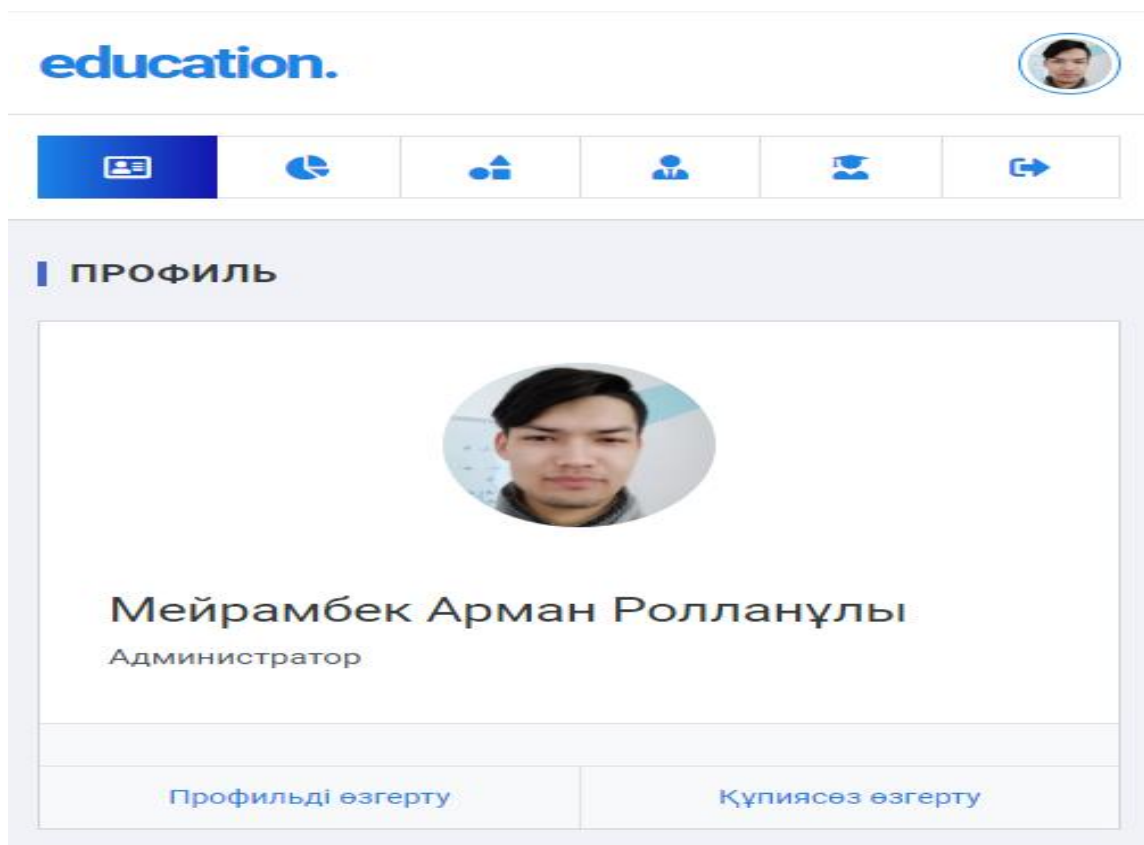
**1.1-сурет – Компьютерлік көрсетілім**

Grid жүйесінің планшеттегі көрсетілімі 1.2-суретінде көрсетілген.



**1.2-сурет – Планшеттік көрсетілім**

Grid жүйесінің мобильдегі көрсетілімі 1.3-суретінде көрсетілген.



1.3-сурет – Мобильдік көрсетілім

## 2.2 JavaScript программалау тілі

JavaScript – көп парадигмалық бағдарламалау тілі. Объектілі-бағытталған, императивті және функционалды стильдерді қолдайды. Бұл ECMAScript-ды енгізу болып табылады.

JavaScript әдетте бағдарлама объектілеріне бағдарламалық қол жеткізу үшін ендірілетін тіл ретінде пайдаланылады. Бұл браузерлерде веб-беттерді интерактивті ету үшін сценарий тілі ретінде кеңінен қолданылады.

Негізгі архитектуралық ерекшеліктер: динамикалық теру, әлсіз теру, автоматты жады басқару, прототипті бағдарламалау, бірінші сыныпты нысандар ретінде жұмыс істейді.

JavaScript-ге көптеген тілдер әсер етті, ал дамыған кезде тілдің Java-ға ұқсастығын, бірақ программалық емес бағдарламаларды пайдалану оңай болды, олар:

– HTML – веб-мазмұнды визуалды және семантикалық түрде құрастырады. Мысалы: Параграфтарды, тақырыптарды, деректер кестелерін немесе беттегі суреттер мен бейнелерді кірістіретін белгілеу тілі;

– CSS – біздің HTML мазмұнын көрсету стилің беретін стиль тілі. Мысалы: Фонға және шрифтке түс қосамыз, мазмұнға көп бағанды көрініс береміз;

– JavaScript – динамикалық жаңартылған мазмұнды жасауға мүмкіндік беретін бағдарламалау тілі. Ол мультимедианы басқарады, бейнелерді анимациялайды және т.б. Javascript коды арқылы көп нәрсеге қол жеткізе алуға болады.

HTML, CSS, JavaScript құрылымы және байланыстары 1.4-суретінде көрсетілген.



```
HTML 7 unsaved changes
1 <p>Player 1: Chris</p>

CSS
1 p {
2   font-family: 'helvetica neue', helvetica, sans-serif;
3   letter-spacing: 1px;
4   text-transform: uppercase;
5   text-align: center;
6   border: 2px solid rgba(0,0,200,0.6);
7   background: rgba(0,0,200,0.3);
8   color: rgba(0,0,200,0.6);
9   box-shadow: 1px 1px 2px rgba(0,0,200,0.4);
10  border-radius: 10px;
11  padding: 3px 10px;
12  display: inline-block;
13  cursor:pointer;
14 }

JS
1 var para = document.querySelector('p');
2
3 para.addEventListener('click', updateName);
4
5 function updateName() {
6   var name = prompt('Enter a new name');
7   para.textContent = 'Player 1: ' + name;
8 }
9
```

PLAYER 1: CHRIS

### 1.4-сурет – JavaScript құрылымы

JavaScript-тың жұмыс істеп тұрған жері 1.5-суретінде көрсетілген.



### 1.5-сурет – JavaScript жұмыс істеп тұрған кезі

JavaScript жауабы 1.6-суретінде көрсетілген.

PLAYER 1: ARMAN

## 1.6-сурет – JavaScript өзгерісі

### 2.2.1 jQuery

jQuery – дайын функцияларды қамтитын JavaScript кітапханасы, барлық jQuery әрекеттері JavaScript кодынан орындалады.

jQuery кітапханасы DOM мазмұнын оңай басқаруға көмектеседі. Сонымен қатар, HTML және CSS-дің бастапқы файлдары өзгермейді, өзгертулер тек пайдаланушыға арналған бетті көрсету үшін жасалады. jQuery кітапханасы AJAX-мен жұмыс істеу үшін қолайлы API-ты ұсынады.

jQuery әдісі веб-беттің мазмұнын басқаруға мүмкіндік береді. Әрбір әдіс параметрді қайтарады немесе қабылдайды және көрсетілгенді орындайды.

Әдісті шақырудың синтаксисі келесідей:

- `$ («селектор») NameMethod (params);`
- бетке мазмұн қосу;
- элементтерді қосу;
- элементтерді ауыстыру және жою;
- класс қосу және жою;
- элементтердің атрибуттарын өзгерту;
- элементтер қасиеттерін өзгерту;
- элементтің мөлшерін және координаттарын қабылдау және өзгерту;
- DOM-ны жылжыту және HTML элементтерін алу;
- әдіс арқылы өз әсерлеріңізді жасау `.animate ()`;
- jQuery анимация эффектілері;
- анимация кезегін басқару;
- jQuery объектінің қасиеттері арқылы анимацияны басқару;
- терезе нысанының әдістері.

jQuery оқиғалары қарапайым пайдаланушы әрекеттеріне жауап беру арқылы веб-беттерді интерактивті жасауға мүмкіндік береді.

Оқиғаның басталуы – оқиға орын алған сәт. Оқиғалар веб-парақпен әр түрлі әрекеттерді орындағанда іске қосылуы мүмкін. Браузер оқиғалардың көзі болуы мүмкін.

jQuery селекторлары веб-беттің элементтерін таңдайды және әдістер осы элементтердегі манипуляцияларды орындайды:

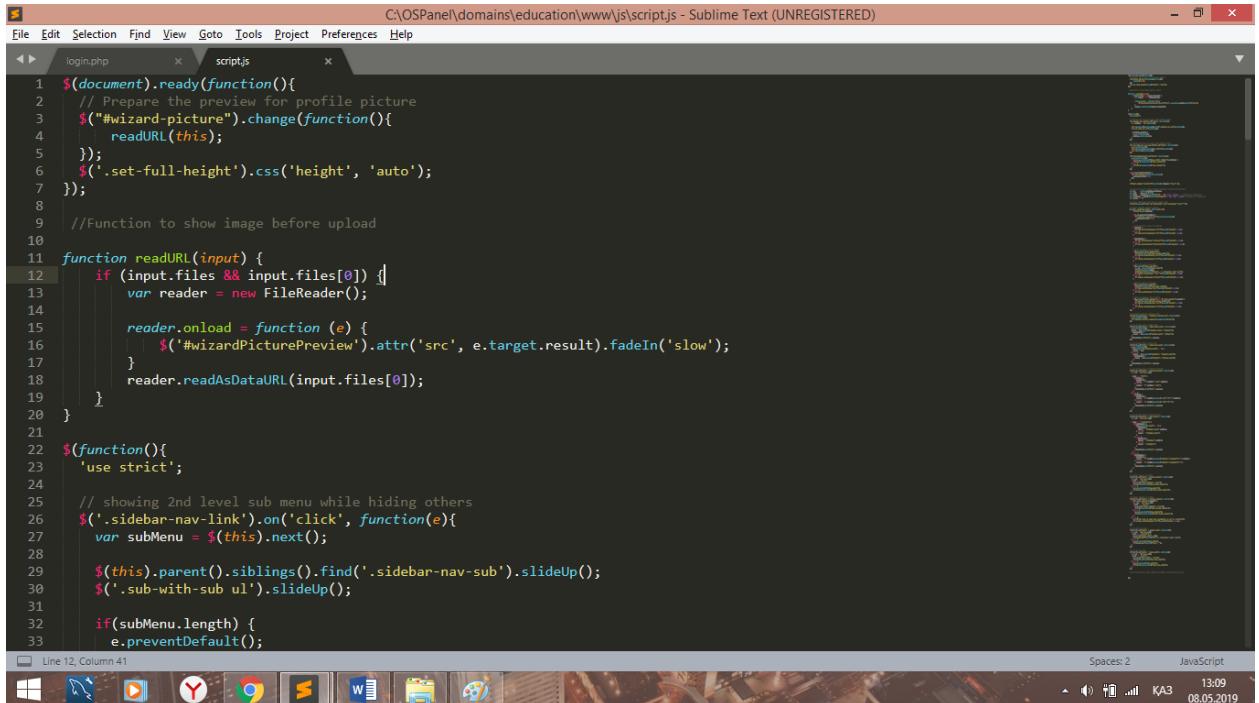
- біріктірілген селекторлар;



- атрибуттарды таңдау;
- сүзгілер.

Бетке қосу үшін келесі кодты бетке қосу керек:<head><script src= "../jquery-2.2.2.min.js"></script></head>

Проектыдағы jQuery-коды(script.js) қолданысы. 1.7-суретінде көрсетілген.



```
1 $(document).ready(function(){
2 // Prepare the preview for profile picture
3 $('#wizard-picture').change(function(){
4 readURL(this);
5 });
6 $('#set-full-height').css('height', 'auto');
7 });
8
9 //Function to show image before upload
10
11 function readURL(input) {
12 if (input.files && input.files[0]) {
13 var reader = new FileReader();
14
15 reader.onload = function (e) {
16 $('#wizardPicturePreview').attr('src', e.target.result).fadeIn('slow');
17 }
18 reader.readAsDataURL(input.files[0]);
19 }
20 }
21
22 $(function(){
23 'use strict';
24
25 // showing 2nd level sub menu while hiding others
26 $('.sidebar-nav-link').on('click', function(e){
27 var subMenu = $(this).next();
28
29 $(this).parent().siblings().find('.sidebar-nav-sub').slideUp();
30 $(subMenu).slideDown();
31
32 if(subMenu.length) {
33 e.preventDefault();
34 }
```

1.7-сурет – jQuery көрсетілімі

## 2.2.2 Ajax

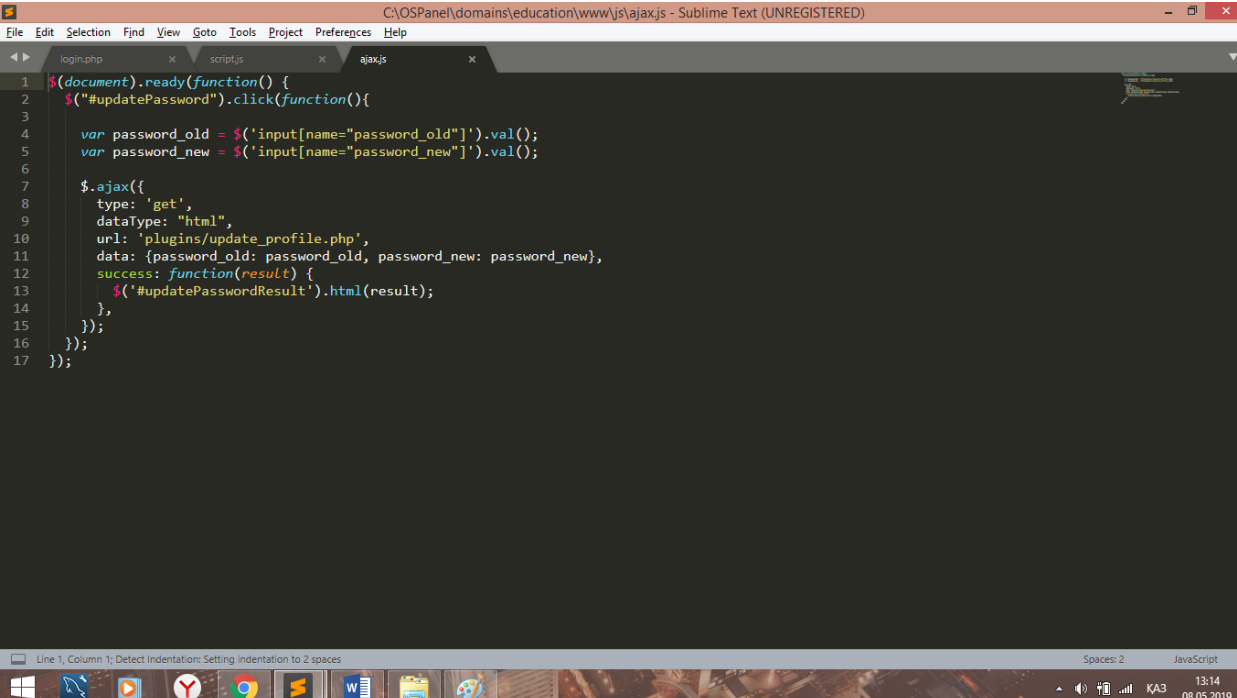
AJAX – Asynchronous Javascript және XML аббревиатурасы. Шындығында, AJAX жаңа технология емес, себебі Javascript және XML бірнеше уақытқа созылды, ал AJAX – белгілі технологиялардың синтезі. AJAX көбінесе Web 2.0 терминімен байланысты және соңғы веб-қосымша ретінде ұсынылады.

AJAX пайдаланған кезде, беттің толық бетін жаңартуға қажеті жоқ, себебі оның нақты бөлігі жаңартылады. Бұл әлдеқайда ыңғайлы, өйткені ол ұзақ күте алмайды және үнемді, өйткені әркімнің шексіз интернеті жоқ. Бұл жағдайда, әзірлеуші бетінде не болып жатқанын пайдаланушы білуі керек. Бұл жүктеу индикаторлары, деректермен сервермен байланысқан мәтіндік хабарламалар арқылы іске асырылуы мүмкін. Сондай-ақ, барлық браузерлер AJAX (браузерлердің және мәтіндік шолғыштардың ескі нұсқаларын) қолдайтынын түсінбеу керек. Плюс Javascript пайдаланушыдан өшірілуі мүмкін. Сондықтан, технологияны пайдалануды теріс пайдаланбауға және веб-сайтқа ақпарат берудің баламалы әдістеріне жүгінуге болмайды.

AJAX артықшылықтары:

- ыңғайлы веб-интерфейсті құру мүмкіндігі;
- белсенді пайдаланушы әрекеттестігі;
- толық емес, толық емес бет жүктеу;
- пайдалану мүмкіндігі.

AJAX веб-парақпен жұмыс істеудің екі әдісін қолданады: веб-бетті қайта жүктеместен өзгерту және серверге динамикалық қол жеткізу. Екіншісі бірнеше тәсілмен жасалуы мүмкін, атап айтқанда XMLHttpRequest. Проектіндегі ajax.js қолданысы 1.8-суретінде көрсетілген.



```
1 $(document).ready(function() {
2   $("#updatePassword").click(function(){
3
4     var password_old = $('input[name="password_old"]').val();
5     var password_new = $('input[name="password_new"]').val();
6
7     $.ajax({
8       type: 'get',
9       dataType: "html",
10      url: 'plugins/update_profile.php',
11      data: {password_old: password_old, password_new: password_new},
12      success: function(result) {
13        $('#updatePasswordResult').html(result);
14      },
15    });
16  });
17 });
```

1.8-сурет – Ajax көрсетілімі

### 2.3 PHP скрипті бағдарламалау тілі

PHP (Hypertext Preprocessor – гипермәтіндік препроцессор) – бұл кеңінен қолданылатын ашық бастапқы коды жалпы мақсаттағы бағдарламалау тілі болып табылады.

PHP – веб-серверде жұмыс істейтін веб-қосымшаларды жазу үшін арнайы әзірленген бағдарламалау тілі. Тілдің синтаксисі көбінесе C, Java және Perl синтаксисіне негізделген. Бұл C мен Perl-ге өте ұқсас, сондықтан кәсіби бағдарламашыға үйрену оңай. Екінші жағынан, PHP тілінің C тіліне қарағанда қарапайым және оны басқа бағдарламалау тілдерің білмейтін веб-мастер оны тез меңгеруі мүмкін. Оны оқып-үйрену қиын емес, себебі оны алдымен байыпты түрде қабылдауға шешім қабылдаған адамдарға көрінуі мүмкін. Бірнеше қарапайым функцияларды зерттегеннен кейін, оларды сіздің веб-

сайтыңызда қолдануға болады, осылайша сіздің жобаңыздың функционалдығын айтарлықтай арттырады.

PHP тілі HTML тілімен бірдей құжатта өмір сүруі мүмкін, сондықтан PHP кодын жазылған HTML – кодқа ендіруге болады. PHP тілінің бұл мүмкіндіктері веб-мастерге өз жұмыстарындағы ең үлкен еркіндікке қол жеткізуге мүмкіндік береді. «Динамика қосу» деген термин нені білдіреді? Мүмкін, бұл сайтта өзгеріс мүмкіндігі, яғни пайдаланушы сұрайтынына байланысты. Іздестіру формасында қандай да бір ақпаратты сұрайды ма, әлде мысалы, қандай да бір есептеулерді орындағысы келеді – бұл әрекеттер PHP тілі арқылы мүмкін болады.

Қысқаша айтқанда, PHP сайтқа функционалдылық қосу үшін пайдаланылады. Жалғыз HTML-ды пайдалана отырып, сіз ешқашан мұндай нәтижеге қол жеткізе алмайсыз!

PHP артықшылығы, мысалы, JavaScript сияқты, PHP сценарийлері сервер жағында орындалады. PHP пайдаланушы компьютерінің немесе оның браузерінің жылдамдығына байланысты емес, ол серверде толығымен жұмыс істейді. Пайдаланушы қалыпты HTML файлы немесе сценарийдің нәтижесін алғанын біле бермейді.

PHP сценарийлері серверде жекелеген файлдар ретінде орындалуы мүмкін немесе олар html беттеріне біріктірілуі мүмкін.

PHP тек HTML құжаттарын ғана емес, түрлі форматтағы суреттерді - JPEG, GIF, PNG, PDF және FLASH файлдарын жасап және түрлендіре алады. PHP XHTML және XML сияқты кез келген мәтіндік форматта деректерді шығара алады.

PHP – кросс-платформа технологиясы. PHP тарату көптеген операциялық жүйелерде, соның ішінде Linux, Unix, Microsoft Windows, Mac OS және көптеген басқа да көптеген өзгерістер үшін қол жетімді. PHP көптеген Apache, Microsoft Internet Information Server (IIS), Microsoft жеке веб-сервері және басқалары сияқты веб-серверлерде қолдау көрсетіледі.

Көптеген серверлер үшін PHP 2-ші нұсқада – модуль ретінде және CGI препроцессоры ретінде келеді.

PHP – ODBC және көптеген дерекқорлармен жұмыс істейді: MySQL, MS SQL, Oracle, PostgreSQL, SQLite және т.б.

PHP программалау тілі, әсіресе танымал MySQL дерекқорымен бірге әртүрлі қиындықтардың интернет-сайттарын жасаудың ең жақсы мүмкіндігі.

PHP тілі үнемі жетілдірілуде және веб-бағдарламалау тілдері саласында ұзақ уақыт бойы басымдықты қамтамасыз етілді.

Мына құрылып жатқан сайттағы PHP – кодын төмендегі index.php(админ беті) 1.9-суретінде көрсетілген.

```

13
14
15 </head>
16 <body>
17
18 <?php
19 session_start();
20 require("config.php");
21
22 if(isset($_SESSION['login_education'])){
23
24     $query_user_info = mysql_query("SELECT * FROM db_users u, db_users_type t
25                                     WHERE u.login = '$_SESSION[login_education]'
26                                     AND t.user_type = u.user_type ");
27     $user_info       = mysql_fetch_array($query_user_info);
28
29     require "pages/modules/header.php";
30     require "pages/modules/main.php";
31     require "pages/content.php";
32
33 }else header('Location:login.php');
34 ?>
35
36 <script src="lib/jquery/js/jquery.js"></script>
37 <script src="lib/popper.js/js/popper.js"></script>
38 <script src="lib/bootstrap/js/bootstrap.js"></script>
39 <script src="lib/jquery.cookie/js/jquery.cookie.js"></script>
40
41 <script src="js/script.js"></script>
42 <script src="js/ajax.js"></script>
43 </body>
44 </html>

```

### 1.9-сурет – Админ бетінің құрылымы

Кіру беті(login.php) 1.10-суретінде көрсетілген.

```

21 <?php
22 session_start();
23 require("config.php");
24
25 if(isset($_POST['log_in'])){
26
27     $login = $_POST['login'];
28     $pass  = $_POST['password'];
29
30     $login = stripslashes($login);
31     $login = htmlspecialchars($login);
32     $login = trim($login);
33     $pass  = stripslashes($pass);
34     $pass  = htmlspecialchars($pass);
35     $pass  = trim($pass);
36
37     $password = md5($pass);
38
39     $user = mysql_query("SELECT * FROM db_users
40                         WHERE login = '$login'
41                         AND password = '$password' ");
42     $row  = mysql_fetch_array($user);
43
44     if($row['user_id'] != ''){
45         $_SESSION['login_education'] = $row['login'];
46         $_SESSION['user_type'] = $row['user_type'];
47         header("Location: ./");
48     }
49     else{
50         echo "<p class='text-danger'> Логин немесе құпиясөз қате! </p>";
51     }
52 }
53 ?>

```

### 1.10-сурет – Кіру бетінің құрылымы

Сонымен қатар РНР математикалық есептеулердің барлық түрлерін – қосу, түсіру, көбейту, бүгінгі күнді, аптаның күнін және жылын анықтауға бөлуді жүзеге асыра алады. РНР пайдаланушы ақпаратын сақтай алады. Яғни, пайдаланушы тікелей сценариймен өзара әрекеттесе алады, мысалы, деректерді контакт пішініне немесе мекенжай кітабына енгізіңіз, іздеу формасы арқылы

сұрауды орнатыңыз, мақалаға түсініктеме жазыңыз, форумда жаңа хабарламалар жасаңыз және т.б.

## 2.4 phpMyAdmin веб-қосымшасы

phpMyAdmin – қарапайым, ыңғайлы және жақсы құжатталған шешім, ол көптеген тілдерге, соның ішінде орыс тіліне аударылады. Пайдаланушы осы бағдарламамен жұмыс істеу туралы сұрақтар туындаса да, интернетте көптеген стандартты сұрақтарға жауап таба аласыз. Phpmyadmin PHP-те жазылған.

Phpmyadmin не істей алады:

- деректер базасын, кестелерді, жазбаларды жасау және реттеу;
- пайдаланушыларды жасау;
- SQL командаларын орындау мүмкіндігі;
- деректер базасын іздеу жүйесі.

Жалпы, дерекқормен жұмыс істеу үшін қажетті барлық қажетті құралдар бар.

Деректер базасымен жұмыс істеу үшін барлық хостерлер өздерінің басқару панелдерінде phpmyadmin орнатады. Бұл қосымша осы класстағы басқа бағдарламалар арасында ең танымал. Бірақ сіз білетін болсаңыз, балама шешімдер бар.

MySQL дерекқорымен қатынасуға және жұмыс істеуге бірнеше тәсілдері бар:

– пәрмен жолын пайдалану. Бұл әдіс пайдаланушыдан кәсіби деңгей мен белгілі бір білім деңгейін талап етеді. Пәрмен жолымен жұмыс істеу үшін тиісті командалар мен ережелерді білуіңіз керек;

– графикалық интерфейс арқылы дерекқормен жұмыс істеуге мүмкіндік беретін веб-қосымшалар.

Бұл әдіс қарапайым пайдаланушыларға ыңғайлы және интуитивті интерфейске ие. Тіпті адам дерекқормен ешқашан жұмыс істемесе де, кез-келген жағдайда ол бірнеше негізгі операцияларды орындай алады.

MySQL деректер базасымен жұмыс істеу үшін осындай веб-қосымшалар – phpmyadmin.

## 2.5 Apache HTTP сервері

Apache HTTP сервері – кросс-платформалық бағдарламалық жасақтама болып табылатын еркін веб-сервер деп аталады. Apache келесі операциялық жүйелерді қолдайды: BSD, Microsoft Windows, Linux, Mac OS, BeOS, Novell NetWare.

Конфигурацияның сенімділігі мен икемділігі Apache-нің басты артықшылығы болып табылады. Бұл бағдарламалық жасақтама арқасында деректерді беру, қате туралы хабарламаларды өзгерту, пайдаланушылардың түпнұсқасын растау үшін ДББЖ пайдалану үшін сыртқы модульдерді қосуға болады. Оны Apache IPv6-ді қолдайды.

1996 жылдың сәуір айынан бастап қазіргі уақытқа дейін Apache – Интернеттегі ең танымал НТТР сервері. Статистикаға сәйкес, 2007 жылдың тамызында НТТР сервері барлық веб-серверлердің 51% жұмыс істеді, 2009 жылғы мамырда бұл көрсеткіш 46%-ға дейін төмендеді және 2011 жылғы қаңтарда ол 59%-ға дейін өсті. Бүгінгі күні веб-сайттардың жалпы санының 59%-дан астамы Apache веб-сервері арқылы беріледі. Apache бағдарламасы Apache Software Foundation компаниясының қолдауымен ашық көздер қауымдастығының сарапшылары әзірлейді және қолдайды. Apache көптеген бағдарламалық өнімдерде, соның ішінде IBM WebSphere және Oracle DBMS.

Apache кірістірілген виртуалды хост механизмімен жабдықталған. Осының арқасында сіз өзіңіздің әрқайсыңыз үшін өзіңіздің контентіңізді көрсете отырып, бір IP-адреске көптеген веб-жобаларға (домендік атауларға) қызмет көрсете аласыз. Әрбір виртуалды хост үшін модульдер мен ядро үшін өздерінің параметрлерін көрсетуге болады, сондай-ақ бүкіл торапқа немесе белгілі бір файлдарға кіруге шектеулер орнатуға болады. Әрбір виртуалды хост үшін Apache – ІТК пайдалану арқылы httpd үдірісін gid және uid идентификаторлары арқылы іске қоса аласыз. Сондай-ақ, әрбір виртуалды хост үшін сервер ресурстарын (трафик, RAM, процессор) шектеу және есепке алу мүмкіндігін беретін модульдер бар.

Apache-ге даму жүйелеріне және көптеген бағдарламалау тілдеріне қолдау көрсететін көптеген модульдер бар. Олардың ішінде:

- Python (mod\_wsgi, mod\_python);
- PHP (mod\_php);
- Perl (mod\_perl);
- Ruby (apache-ruby);
- ASP (apache-asp).

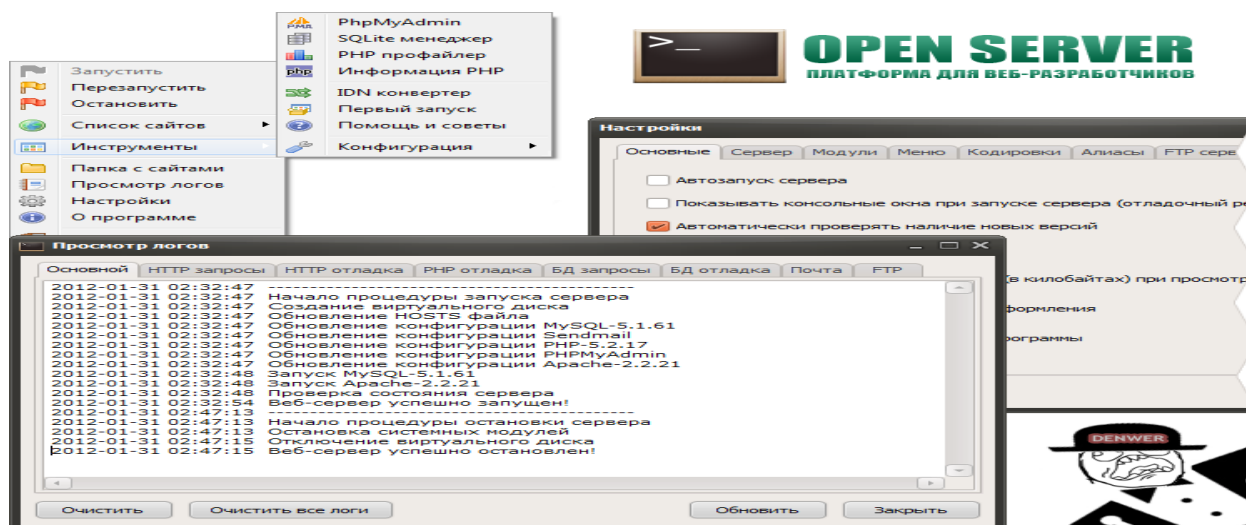
Сонымен қатар, Apache FastCGI және CGI бағдарламаларын қолдана отырып, бағдарламаларды Java, sh, C, C ++ сияқты бағдарламалау тілдерінде іске қосуға мүмкіндік береді.

## 2.6 Open Server

Open Server – көп функционалды басқару бағдарламасы және қосылған компоненттердің үлкен таңдауымен тасымалданатын жергілікті WAMP / WNMP сервері. Ұсынылған бағдарламалық пакет тек «тізеге» жиналған әуесқойлық құрылыс емес, бұл олардың ұсыныстары мен ұсыныстарын ескере



отырып, веб-әзірлеушілерге арналған арнайы толыққанды кәсіби құрал. Open Server 1.11-суретінде көрсетілген.



1.11-сурет – Open Server көрінісі

Әр түрлі орталарда сценарийлерді отладтау үшін Open Server екі түрлі HTTP серверлерін, PHP және DBMS модульдерінің әртүрлі нұсқаларын, сондай-ақ олардың арасында жылдам ауысу мүмкіндігін ұсынады. Олар:

- HTTP модулі: Apache 2.2.21 и Nginx 1.0.11;
- СУБД модулі: MySQL 5.1.61, MySQL 5.5.20 и PostgreSQL 9.1.1;
- PHP модулі: PHP 5.2.17 (IMagick 2.2.1, Zend Optimizer 3.3.3, IonCube Loader 4.0.7, Memcache 2.2.4) и PHP 5.3.9 (IMagick 2.3.0, Xdebug 2.1.3, IonCube Loader 4.0.10, Memcache 2.2.6).

Үздік құралдар жиынтығы: HeidiSQL, Adminer, PHPMyAdmin, PHPPgAdmin, PgAdmin.

Сондай-ақ, пакетте келесідей жақсы өнімдер бар: Perl, FTP сервері, Sendmail, Memcached сервері.

Open Server – бұл Nginx дегенді білетін жалғыз жоба. Мұнда қайта жазу ережелерін ыңғайлы қосу доменнің түбіріндегі .htaccess файлдары арқылы жүзеге асырылады және PHP True FastCGI режимінде жұмыс істейді (өлтірілмейді).

Барлық компоненттер ресми репозиторийлерден алынады және әр жаңарту бумасымен әрқашан ең соңғы нұсқаларға жаңартылады.

## 2.7 MySQL – деректер қоры

MySQL – клиент-сервер моделі бар ашық дереккөзді дерекқорды басқару жүйесі (RDBMS). RDBMS – дерекқорды реляциялық үлгіге негізделген құру

және басқару үшін пайдаланылатын бағдарламалық жасақтама немесе қызмет. Енді әрбір терминді мұқият қарастырайық:

– дерекқор – жай ғана құрылымдық деректер жинағы. Мысалы, өз өзіне селфи жасаған кезде: түймені басып, өзіңізді суретке түсіресіз. Фотосуретіңіз – деректер болып табылады және сіздің телефоныңыздың галереясы – дерекқор болып табылады. Дерекқор – деректер сақталатын орын. «Реляциялық» сөзі деректер жинағында сақталған деректер кестелер түрінде жасалғанын білдіреді. Әрбір кесте қандай да бір түрде байланысты. Егер бағдарламалық жасақтама реляциялық деректер үлгісін қолдамаса, оны жай DBMS деп атаңыз;

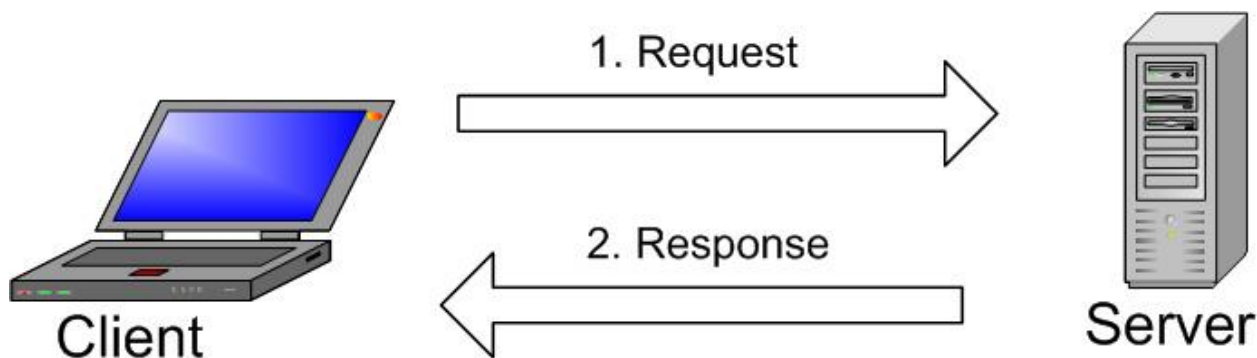
– ашық дереккөз сіз оны қолдануға және өзгертуге құқығыңыз бар екенін білдіреді. Кез келген адам бағдарламалық жасақтаманы орнатуы мүмкін. Сондай-ақ, қажеттіліктеріңізге сай болу үшін бастапқы кодты зерттеуге және теңшеуге болады. Дегенмен, GPL (GPU қоғамдық лицензиясы) шарттарға байланысты не істей алатыныңызды анықтайды. Егер сізге икемді иелену және кеңейтілген қолдау қажет болса, коммерциялық лицензияланған нұсқа қол жетімді;

– RDBMS бағдарламалық жасақтамасын орнататын және басқаратын компьютерлер клиенттер деп аталады. Деректерге кіру қажет болғанда, олар RDBMS серверіне қосылады. Бұл клиент-серверлік жүйе.

MySQL – RDBMS бағдарламалық жасақтамасының көптеген нұсқаларының бірі. MySQL-тың танымал болуына байланысты RDBMS және MySQL-ді бірдей деп санайды. Facebook, Twitter, YouTube, Google және Yahoo сияқты ірі веб-қосымшалар қандай? барлық деректерді сақтау үшін mysql пайдаланады. Бастапқыда шектеулі қолдануға арналған болса да, Linux, macOS, Microsoft Windows және Ubuntu сияқты көптеген маңызды есептеуіш платформаларымен үйлесімді.

MySQL және SQL сәйкес келмейді. MySQL клиент-сервер моделін іске асыратын RDBMS бағдарламалық жасақтамасының ең танымал брендтерінің бірі екенін есте сақтаңыз. Сонымен, клиент пен сервер RDBD ортасында қалай әрекет етеді? Олар доменге тән тілдің құрылымдық сұрау тілін (SQL) пайдаланады. Егер PostgreSQL және Microsoft SQL Server сияқты басқа SQL атауларымен кездескен болсаңыз, олар SQL синтаксисін пайдаланатын брендтер болып табылады. RDBMS бағдарламалық жасақтамасы басқа бағдарламалау тілдерінде жиі жазылады, бірақ әрқашан дерекқормен өзара әрекеттесу үшін SQL тілін негізгі тіл ретінде пайдаланады. MySQL өзі C және C ++-да жазылған. Оңтүстік Америка елдері туралы ойланыңыз, олар географиялық жағынан әртүрлі және басқа тарихы бар, бірақ олар негізінен испан тілінде сөйлейді.

MySQL қалай жұмыс істейтіні 1.12-суретінде көрсетілген.



**1.12-суреті – Базанын көрсетілімі**

Суретте негізгі клиент-сервер құрылымы түсіндіріледі. Бір немесе бірнеше құрылғы (клиенттер) нақты желі арқылы серверге қосылады. Әрбір клиент графикалық пайдаланушы интерфейсінен (GUI) сұрауды өз экрандарында жасай алады және егер екеуінің де екеуі нұсқаулықты түсінсе, сервер қажетті нәтиже береді. Техникалық аспектілерге кірмей, MySQL ортасындағы негізгі процестер бірдей:

- MySQL әр кестенің қатынастарын анықтайтын деректерді сақтау және басқару үшін дерекқорды жасайды;
- клиенттер сұрауларды MySQL-те белгілі бір SQL-пәрмендерін енгізу арқылы жасай алады;
- сервер бағдарламасы сұралған ақпаратпен жауап береді және клиенттік тарапта пайда болады.

MySQL-дің танымалдылығың 1.13-суретінде көрсетілген.

341 systems in ranking, December 2018

Rank			DBMS	Database Model	Score		
Dec 2018	Nov 2018	Dec 2017			Dec 2018	Nov 2018	Dec 2017
1.	1.	1.	Oracle +	Relational DBMS	1283.22	-17.89	-58.32
2.	2.	2.	MySQL +	Relational DBMS	1161.25	+1.36	-156.82
3.	3.	3.	Microsoft SQL Server +	Relational DBMS	1040.34	-11.21	-132.14
4.	4.	4.	PostgreSQL +	Relational DBMS	460.64	+20.39	+75.21
5.	5.	5.	MongoDB +	Document store	378.62	+9.14	+47.85
6.	6.	6.	IBM Db2 +	Relational DBMS	180.75	+0.87	-8.83
7.	7.	↑8.	Redis +	Key-value store	146.83	+2.66	+23.59
8.	8.	↑10.	Elasticsearch +	Search engine	144.70	+1.24	+24.92
9.	9.	↓7.	Microsoft Access	Relational DBMS	139.51	+1.08	+13.63
10.	10.	↑11.	SQLite +	Relational DBMS	123.02	+0.31	+7.82

**1.13-сурет – Базалардың рейтингі**

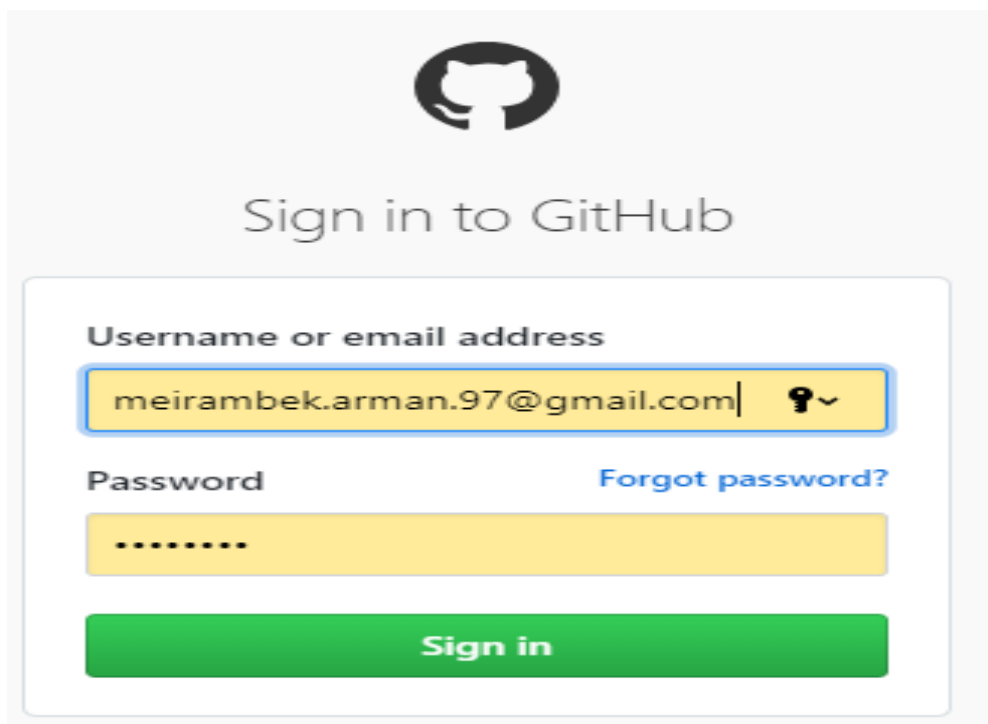
MySQL – бұл нарықтағы жалғыз RDBMS емес, бірақ ол іздеу нәтижелерінде, профильдегі профильдердің саны және Интернет –

форумдардағы техникалық талқылаулардың жиілігі сияқты маңызды параметрлерді пайдаланған кезде Oracle деректер базасына ең танымал және екінші орында. Көптеген ірі технологиялық алыптар оған сенім артып, лайықты орынды нығайтады.

## 2.8 GitHub репозиторий

GitHub – жоба мен код нұсқасын басқару жүйесі, сонымен қатар әзірлеушілерге арналған әлеуметтік желі платформасы. Сонымен қатар, әлемдегі басқа адамдармен жұмыс істеуге, жобаларыңызды жоспарлауға және жұмысыңызды бақылауға мүмкіндік береді.

GitHub-де бүкіл әлем бойынша онлайн-ортақ жұмыс істейтін онлайн-репозиторийлердің бірі болып табылады. GitHub-қа кіру беті 1.14-суретінде көрсетілген.



1.14-сурет – Github-қа кіру беті

Біз жұрместен бұрын, Гиттың не екенін түсінуіміз керек. Бұл гитубтың жүрегі сияқты. Git – нұсқаларды басқару жүйесі. Оны әзірлеген Linus Torvalds (Linux-ды жасаған адам).

Git – GitHub жүрегі болса, Hub – оның жаны. GitHub торабы – Git секілді командалық жолды әзірлеушілер үшін ең үлкен әлеуметтік желіге айналдырады.

Арнайы жобаға қатысудан басқа, GitHub пайдаланушыларға пікірлес адамдармен қарым-қатынас жасауға мүмкіндік береді. Адамдарды қадағалап, не істейтіндерін не олармен байланысатындарын көре аласыз.

Репозиторий – жоба файлдары сақталатын каталог. Ол GitHub репозиторийінде немесе компьютердегі жергілікті репозиторийде орналасуы мүмкін. Код файлдарын, суреттерді, дыбысты немесе репозиторийдегі жобаға қатысты кез келген нәрсені сақтауға болады.

Тармақ сіздің репозиторияңыздың көшірмесі болып табылады. Сіз бөлімді оқшауланғанда жасағыңыз келгенде пайдалануға болады.

Тармақпен жұмыс істеу орталық репозиторийге немесе басқа салаларға әсер етпейді. Егер сіз осы жұмысты орындаған болсаңыз, өз тармағыңызды басқа тармақтармен және орталық қоймамен сұратуды сұрау арқылы біріктіре аласыз.

GIT ерекшеліктерін атап өтуге болады:

- бөлінген нұсқаны басқару жүйесі, GIT клиент-сервер үлгісіне (клиент-сервер) негізделген Subversion (SVN) сияқты басқа жүйелерден айырмашылығы peer to peer қағидаты (теңімен тең).

- GIT әзірлеушілерге көптеген толық тәуелсіз кодтар салуға рұқсат береді. Бұл филиалдар проблемаларсыз және уақытты талап етпестен құруға, жоюға және біріктіруге көмектеседі.

- GIT-де барлық операциялар атомдық болып табылады; Бұл кез-келген әрекеттің толықтай табысты немесе сәтсіз болуы мүмкін дегенді білдіреді (өзгеріссіз). Бұл шынында да маңызды, себебі кейбір әрекеттер атомдық емес, басқарудың барлық жүйелерінде (мысалы, CVS), барлық репозиторийдегі кейбір ілулі операциялар оны тұрақсыз күйде қалдыра алады.

- метадеректер жасырын қалталарда (.cvs, .svn, т.б.) сақталған SVN немесе CVS сияқты басқа VCS-лерден өзгеше, GIT барлық деректер .git каталогтарында орналасқан.

- ол репозиторийде бар барлық нәрсенің криптографиялық тұтастығын қамтамасыз ететін деректер үлгісін пайдаланады. Файлдар қосылған сайын немесе жасалса, олардың жинақталымдары жасалады; Осындай процесс олар шығарылғанда пайда болады.

- GIT-дегі тағы бір тамаша ерекшелігі – оның индексі. Индекс ішіндегі әзірлеушілер тапсырманы пішімдеуге және нақты қолданар алдында оларды көре алады.

Git-ты пайдалану өте оңай. Стартерлер үшін сіз жаңа репозиторийді жасай аласыз немесе барын қосасыз. Орнатқаннан кейін git-init пәрмені сізге жаңа репозиторийді орнатуға көмектеседі немесе git clone пәрмені жергілікті репозиторийдің жұмыс көшірмесіне пайдаланушы орнатуға көмектеседі.

### 3 Жобалау бөлімі

#### 3.1 Rational Rose туралы түсінік

Rational Software Corp. компаниясының объектілі-бағытталған ақпараттық жүйелерін визуалды модельдеуге арналған танымал құрал. Өнім Universal Modeling Language (UML) негізделген. Бірегей модельдік тілдің арқасында Rational Rose ақпараттық жүйелерді жобалаудағы кез-келген мәселені шеше алады: бизнес-процестерді нақты бағдарламалау тілінде кодтауға дейін талдау. Тек Rational Rose жоғары деңгейлі және төмен деңгейлі модельдерді де дамытады, осылайша дерексіз дизайнды немесе логиканы іске асырады.

Rational Rose объектілі-бағытталған дизайн құралы, яғни CASE технологиясын енгізу. CASE – технологиясының тұжырымдамасы компьютерді қолдана отырып, компьютерлік бағдарламалық қамтамасыз ету / жүйелік инженерия – дамушы жүйелер (немесе бағдарламалық қамтамасыз ету) болып табылады.

Rational Rose-те бизнес-процестерді модельдеу түрлі аспектілерді қолдану арқылы жүзеге асырылады. Осы аспектілердің әрқайсысы процестердің нақты сипаттамалары мен мүмкіндіктеріне баса назар аударады.

Бұл аспектілерге мыналар жатады:

– пайдаланыңыз корпусы (Case пайдалану). Бұл аспект процестің қатысушылары қалай әрекет ететінін түсініп, олардың өзара әрекеттесуін және процесіне әсерін анықтайды. Осы аспектісінде технологиялық үлгілерді құру үшін, пайдалану диаграммалары, дәйектеме диаграммалары, ынтымақтастық диаграммалары және әрекет диаграммалары пайдаланылады;

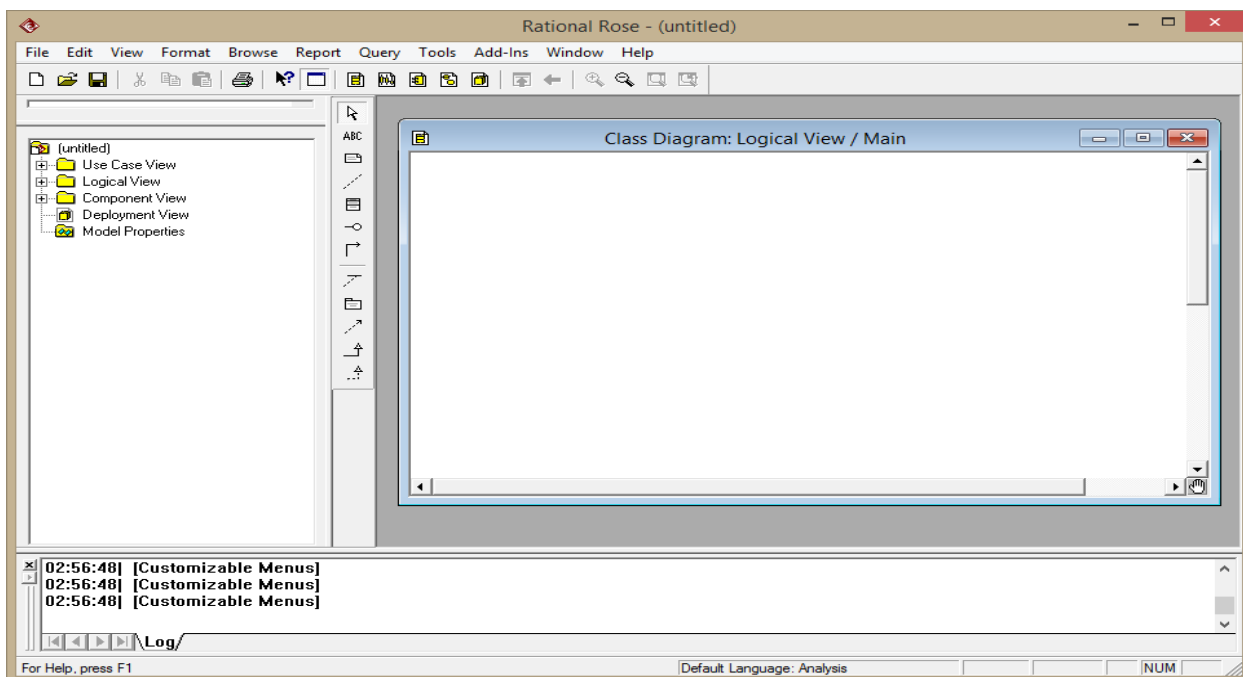
– логикалық аспект Осы аспектіде процестердің функционалдық талаптары анықталуы мүмкін. Ол процесс элементтерінің кластары арасындағы логикалық қатынасты анықтайды. Үлгі диаграммалары мен күй диаграммалары пайдаланылған модельдерді құру;

– құрылымдық элементтер. Бұл аспект процестің элементтерінің құрамына және ақпараттық жүйені құрғанда олардың таралуына назар аударады. Осы аспектілерде модельдер компоненттік схема бойынша құрастырылған. Ол процестің элементтері мен бағдарламалық жасақтамасы туралы ақпаратты қамтиды;

– пайдалануға беру Бұл аспект ақпараттық жүйенің аппараттық құралдарына қатысты технологиялық диаграмманы көрсетеді. Үлгілерді құру үшін тек бір диаграмма қолданылады – топология диаграммасы.

Rational Rose компаниясының ең соңғы нұсқалары бірнеше бизнес-процестерді модельдеуге мүмкіндік беретін бірнеше бағдарламалық өнімдерді қамтиды. Пайдаланушылар бизнес қажеттіліктеріне жақын процестердің графикалық модельдерін жасай алады. Программаның негізгі терезесі 1.15-суретінде көрсетілген.





### 1.15-сурет – Программаның негізгі терезесі

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:

- топтық қолдау. Бұл CASE құралы жобаның барлық қатысушыларына қарапайым қолдауды жүзеге асырды. Пайдаланушылар ортақ үлгілермен қарым-қатынасты сақтай отырып, өздерінің жұмыс орындарын өзгертпестен өздерінің бірегей үлгілері мен өз ортасында жұмыс істей алады;

- моделдерді басқару. Барлық жасалған модельдер оңай өзгеруі мүмкін. Бір үлгідегі өзгерістер өзара байланысты үлгілерде автоматты түрде көрсетіледі. Нұсқаларды басқару жүйесі мен конфигурацияны басқару модельдерін басқару. Бұл бизнес процесінің кез келген үлгілерінде өзгерістер жасауды жеңілдетеді;

- қатені бақылау. Рационалды раушан модельдеу кезінде пайда болатын қадағалау қателерін береді. Бұл қателерді олардың мұрагері бойынша түзетуге және келесі модельдеу деңгейіне көшіруге мүмкіндік береді;

- құжаттау модельдері. Пайдаланушылар процесстік модельдер үшін қажетті есептер мен құжаттарды жасай алады. Құжаттар пайдаланушының мұқтаждықтары үшін жасалады және түрлі үлгілерге қолдануға бейімделуі мүмкін;

- конфигурацияны басқару. Пайдаланушылар интерфейс теңшелімін және бағдарламалардың бөліктерін олардың қажеттіліктерін қанағаттандыру

үшін баптай алады. Rational Rose-де графикалық пайдаланушы интерфейсі (GUI) қолданылады, ол арқылы ыңғайлы жұмыс үшін қажетті ортаны теңшей аласыз.

### 3.1.1 Қолдану(прецедент) диаграммасы

Пайдалану схемалары 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.

Қолдану диаграммасы актердің көзқарасынан жүйеде белгілі бір нәтижеге әкелетін әрекеттер тобын сипаттайды.

Қолдану жағдайлары жүйенің пайдаланушылары мен жүйенің өзі арасындағы типтік өзара әрекеттесу сипаттамалары болып табылады. Олар жүйенің сыртқы интерфейсін көрсетеді және жүйенің не істеуі керек екенін көрсетеді.

Қолдану жағдайларымен жұмыс істеу кезінде бірнеше қарапайым ережелерді есте сақтау маңызды:

- әр қолданылатын жағдай кем дегенде бір актерге қатысты;
- әр жағдайда пайдалану бастамашысы бар;
- әрбір қолдану жағдайы сәйкес нәтижеге әкеледі.

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. Қарапайым ассоциация актриса мен пайдалану жағдайының 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. Оқушылардың қолдану диаграммасы 1.16 – суретінде көрсетілген.



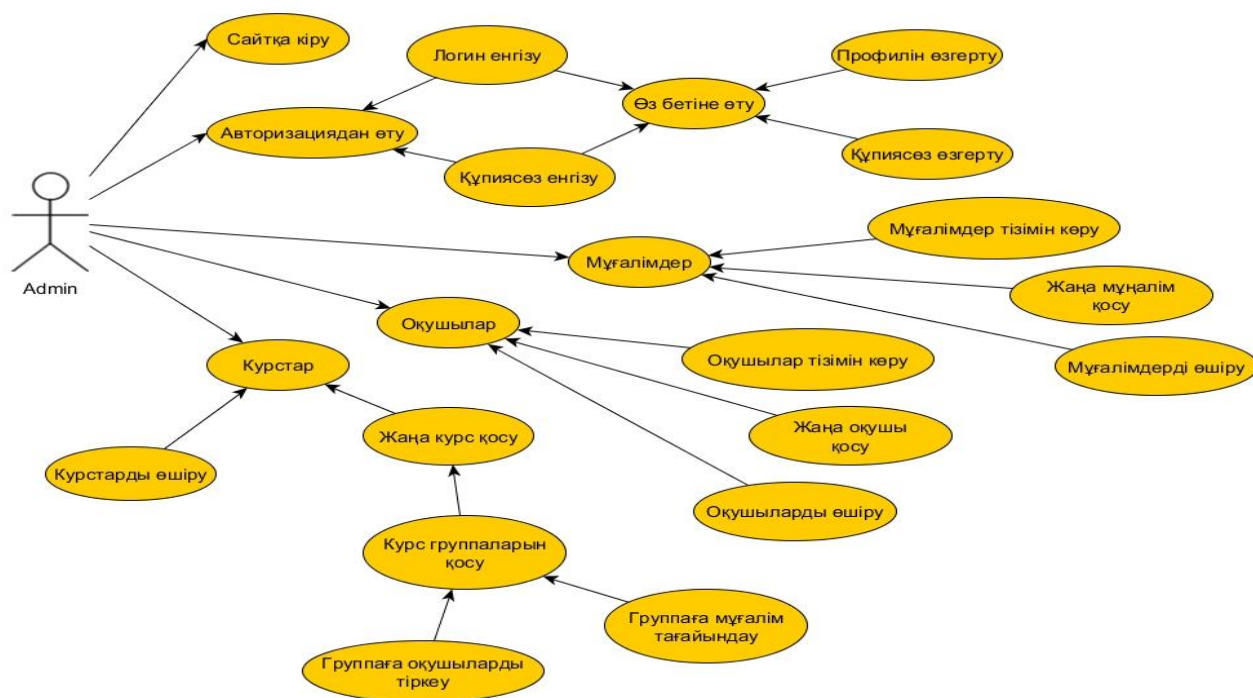
1.16-сурет – Оқушылардың қолдану диаграммасы

Мұғалімдердің қолдану диаграммасы 1.17 – суретінде көрсетілген.



1.17-сурет – Оқушылардың қолдану диаграммасы

Администратордың қолдану диаграммасы 1.18 – суретінде көрсетілген.



**1.18-сурет – Администратордың қолдану диаграммасы**

### 3.1.2 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;

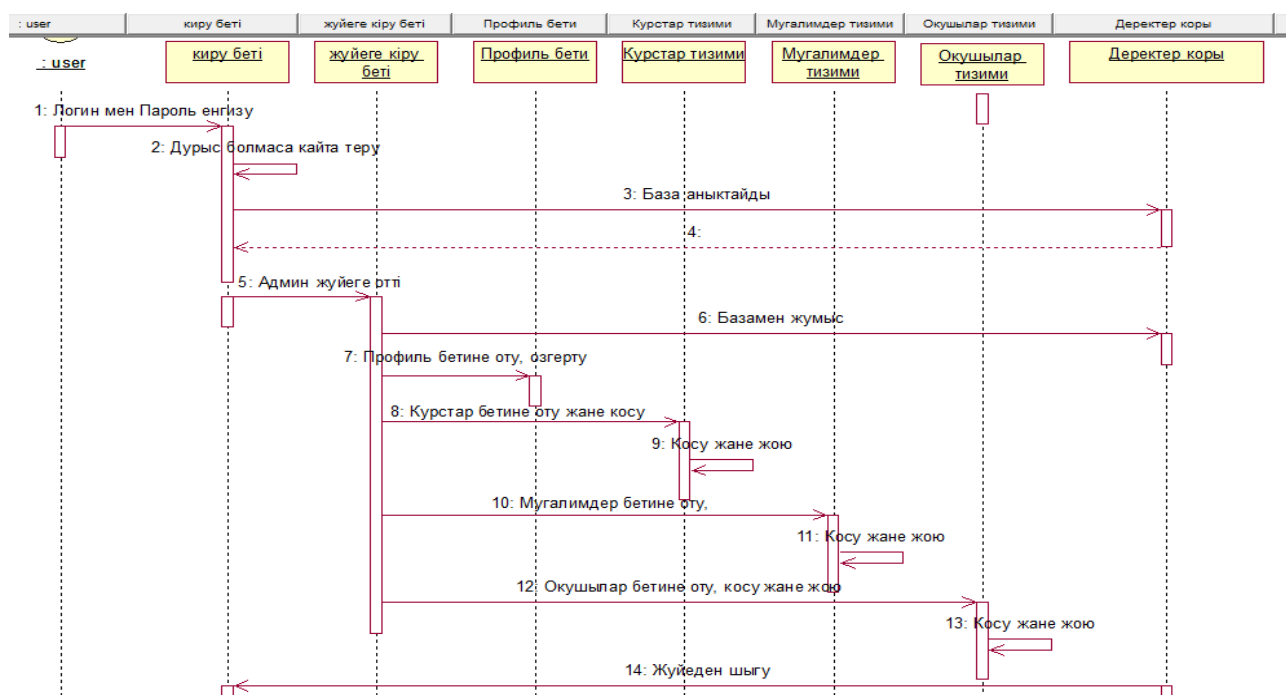
– жауап беру хабарламасы – бұл хабар синхрондық хабарламаға жауап болып табылады. Әдетте, бастапқы жіберуші актерге қайтарылған кез келген мән бар, ол бақылауды қайтарады. (әрекет ету қабілеті);

– асинхрондық хабар – жіберуші актер прецедентте кейбір әрекеттерді жасауы керек қабылдаушы актерге бақылауды жібереді. Синхронды хабардың негізгі айырмашылығы – бұл жіберуші актер басқа әрекеттерді орындау мүмкіндігін жоғалтпайды;

– жоғалған хабарлама – алушының хабарламасы жоқ (жіберуші бар, алушы жоқ);

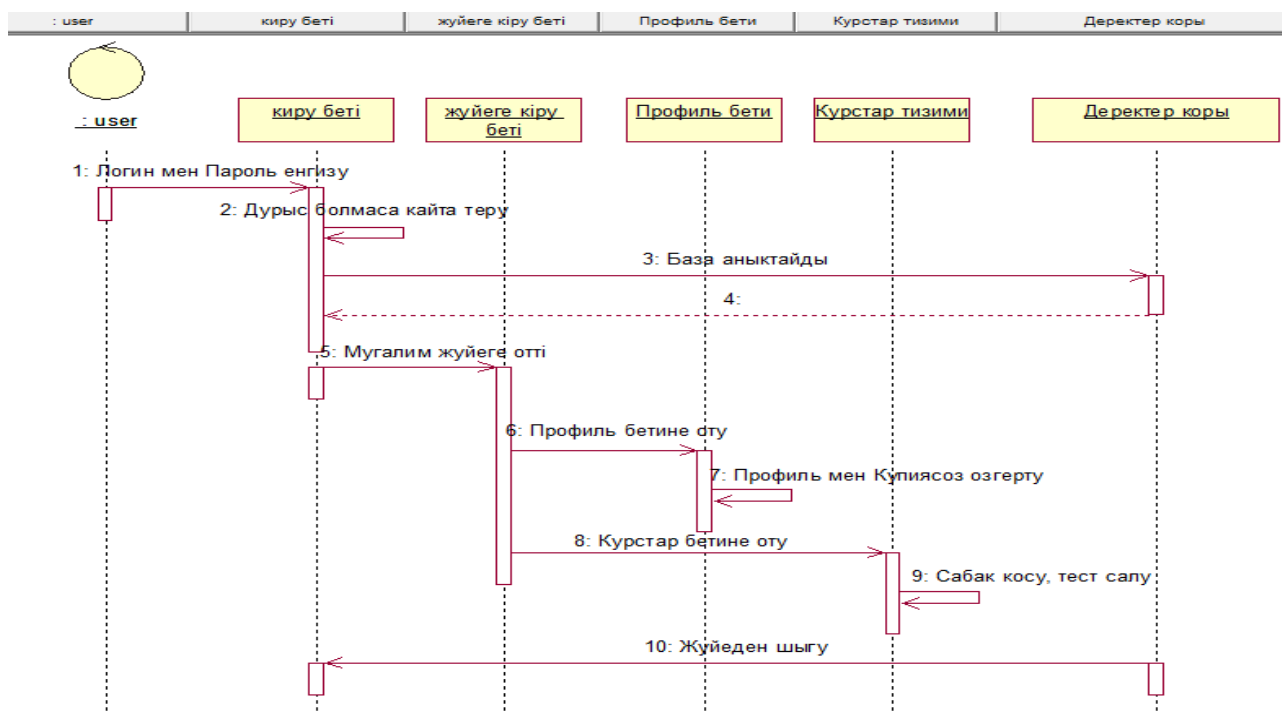
– хабарлама жіберуші жоқ хабар болып табылады.

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. Админның тізбек диаграммасы 1.19 – суретінде көрсетілген.



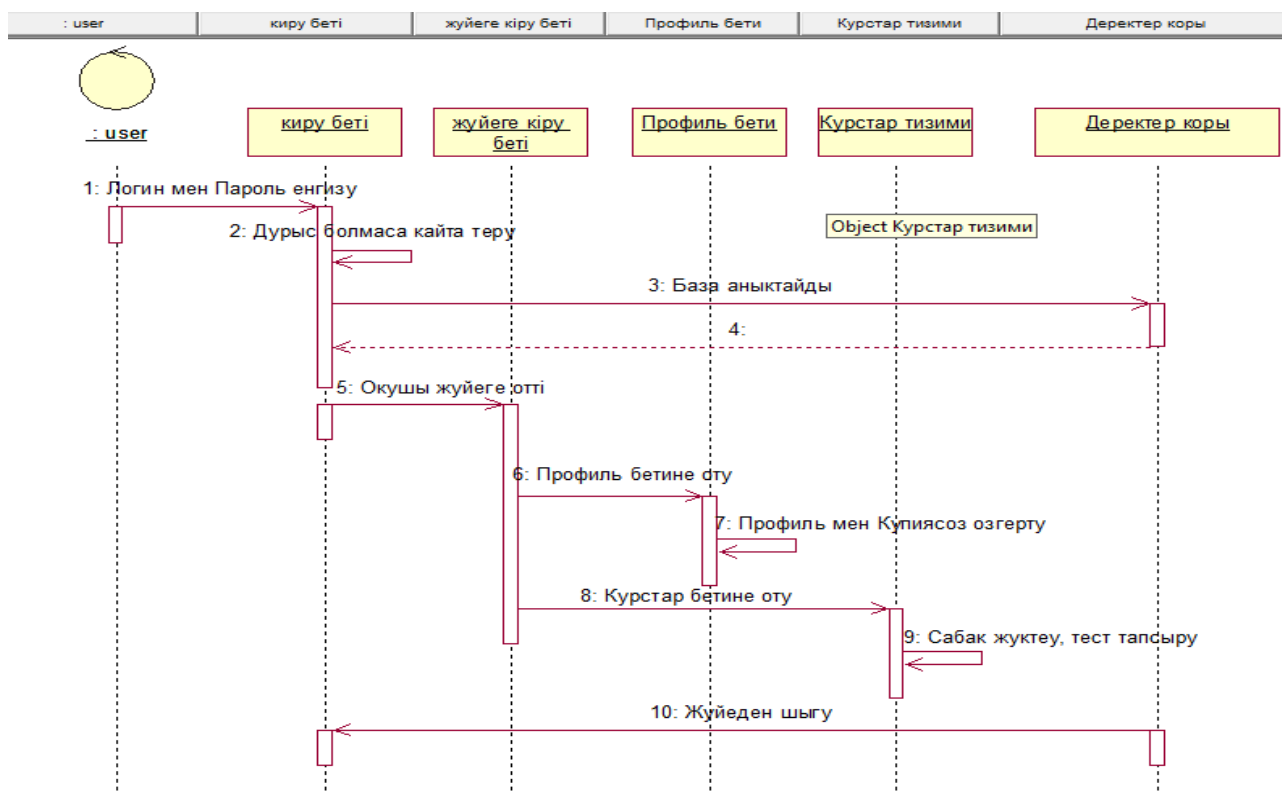
1.19-сурет – Админның тізбек диаграммасы

Мұғалімнің тізбек диаграммасы 1.20 – суретінде көрсетілген.



1.20-сурет – Мұғалімнің тізбек диаграммасы

Оқушының тізбек диаграммасы 1.21 – суретінде көрсетілген.



1.21-сурет – Оқушылардың тізбек диаграммасы



### 3.1.3 Класстар диаграммасы

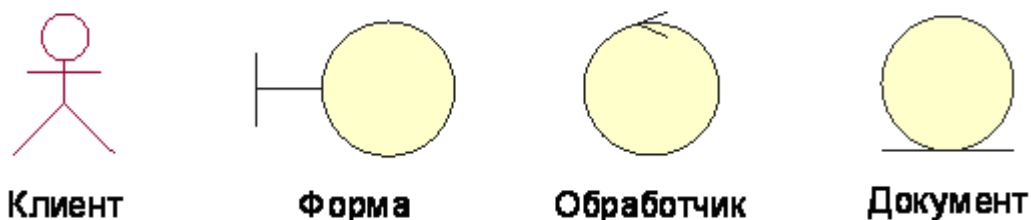
Класс диаграммалары PS моделдеуде жиі қолданылады. Олар жүйенің статикалық сипаттамаларының бірі болып табылады, оның құрылымын көрсететін оның дизайны тұрғысынан. Класс диаграммасы онда көрсетілген класстардың нысандарының динамикалық әрекетін көрсетпейді. Класстық диаграммалар класстарды, интерфейстер және олардың арасындағы қатынастарды көрсетеді.

Класс – PS негізгі блок. Бұл тұжырым OO бағдарламалау тілдерінде де бар, яғни UML сыныптары мен бағдарламалық жасақтама сыныптары арасында сәйкестік бар, бұл бағдарлама кодтарын автоматты түрде жасау немесе қайта құру үшін негіз болып табылады. Әрбір сыныпта атау, атрибуттар және операциялар бар. Диаграммадағы сынып үш бағытқа бөлінген тіктөртбұрыш ретінде көрсетіледі. Жоғарғы сынып атауы, ортасында атрибуттардың сипаттамасы бар (қасиеттері), төменгі бөлігінде осы сыныптың нысандарымен қамтамасыз етілген операциялар мен қызметтердің атаулары бар.

Әрбір класс төлсипаты үшін көрінуді орнатуға болады. Бұл сипат атрибут басқа класстар үшін қол жетімді екенін көрсетеді. UML келесі төлсипаттардың көріну деңгейлерін анықтайды:

- ашық (жалпыға ортақ) – атрибут кез келген басқа сынып үшін (объект) көрінеді;
- қорғалған – атрибут осы сыныптың ұрпақтары үшін көрінеді;
- жеке – атрибут сыртқы сыныптарға (нысандарға) көрінбейді және оны ғана қамтитын нысанмен пайдалануға болады.

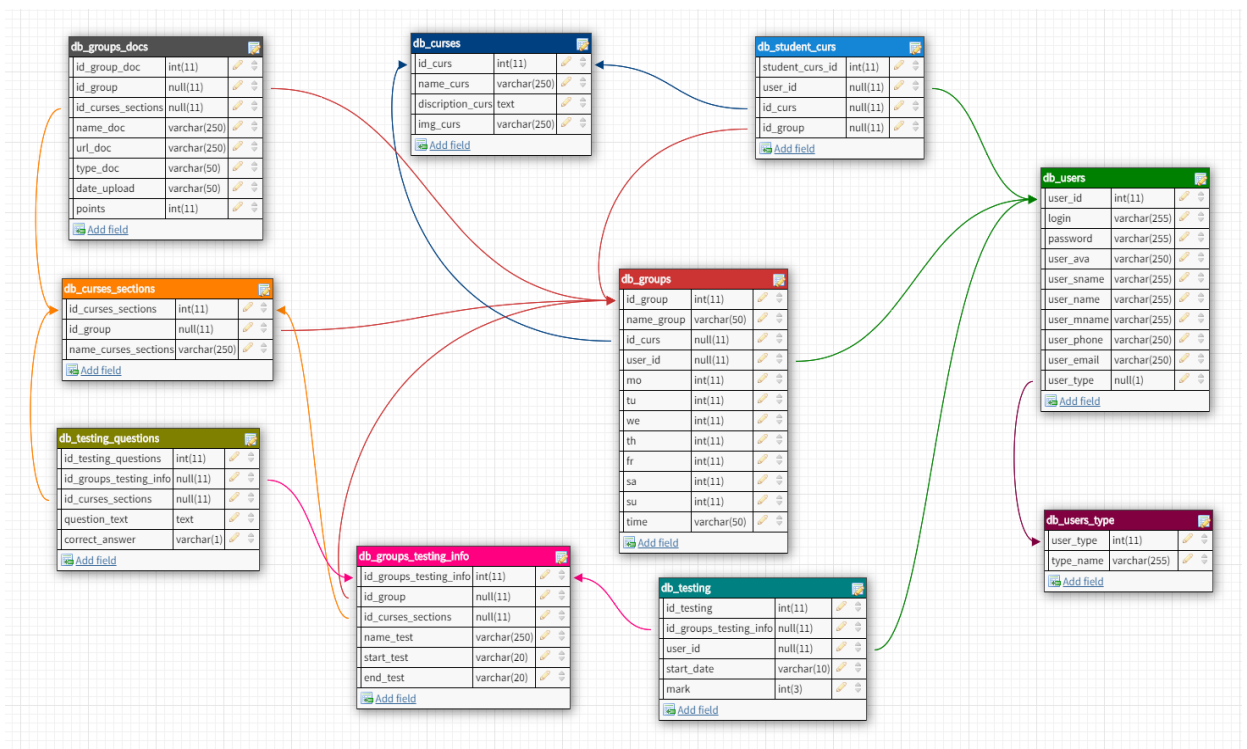
Стереотип класс – субъектілерді (деректер класстары) белгілеу үшін пайдаланылады, стереотип PS пен оның сыртындағы субъектілер арасындағы делдалдар болып табылатын шекаралық класстарды сипаттайды – стереотиптен тұратын актерлер. Ақырында, стереотипте өзара әрекеттесуді басқаратын класстар мен объектілер сипатталады. Стереотиптерді пайдалану, атап айтқанда, класс диаграммасының түрін өзгертуге мүмкіндік береді. Стереотиптердің 1.18-суретінде көрсетілген.



1.18-сурет – Класстарға арналған стереотиптер

Класс диаграммалары PS-дің логикалық симуляциясында жасалады және келесі мақсаттарда қолданылады:

- PS сәулетін көрсету. Архитеуралық маңызы бар класстарды сәйкестендіруге және PS архитектурасын сипаттайтын диаграммаларда көрсетуге болады;
  - Навигация экрандарын имитациялау үшін. Бұл диаграммалар шекаралық сыныптарды және олардың логикалық қатынастарын көрсетеді. Ақпараттық өрістер сыныптық атрибуттар, ал басқару түймелері әрекеттер мен қатынастар ретінде модельденеді;
  - бағдарламалық компоненттердің логикасын модельдеу (келесі мақалаларда сипатталған);
  - деректерді өңдеу логикасын моделдеу.
- Кластар диаграммасы 1.19 – суретінде көрсетілген.



1.19-сурет – Класстар диаграммасы

### 3.1.4 Қызмет диаграммасы

Қызмет диаграммасы – жүйе динамикасын модельдейтін UML-дің бас диаграммаларының бірі. Қызмет диаграммасы күйден күйге ауысатын басқару ағынын айқындай отырып, шеткі автоматты бейнелейді. Шеткі автомат – бұл объекттің бар болуы барысында қызмет тізбегін анықтайтын тәртіп.

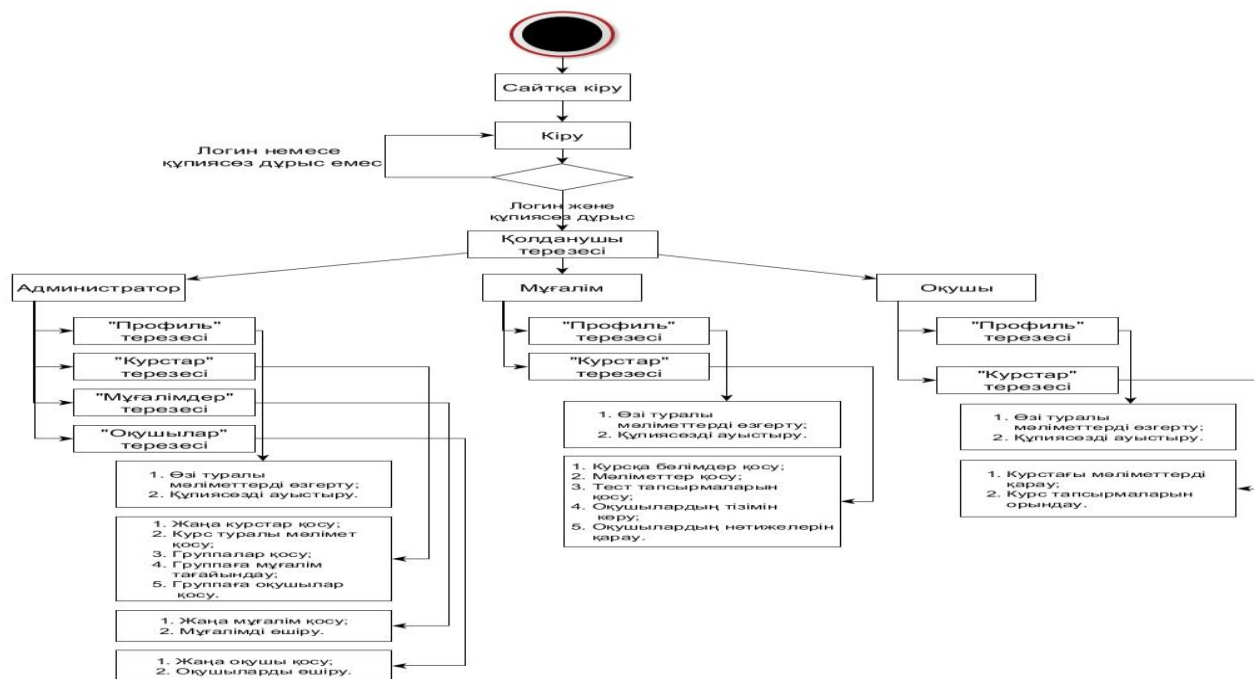
Қызмет диаграммасы келесіні көрсетеді:

- жүйе қызметінің жиыны;
- бір қызметтен екінші қызметке көшуді шақыратын қызмет;
- қызмет өзгерісі нәтижесінде болатын әрекеттер.

UML тілінде қызмет деп объектінің өміріндегі аралықты атайды, ол бұл кезде қандай да бір шартты қанағаттандырады, белгілі бір іскерлікті белгілі бір оқиғаны күтеді.

Оқиға – қызмет өзгерісіне әкелетін жағдай. Әрекет – оқиға іске қосатын операциялар жиыны.

Қызмет диаграммалары реактивті жүйелерді модельдеу кезінде өте пайдалы оқиғалармен басқарылатын объект тәртібін айқындайды. Қызмет диаграммасы 1.20 – суретінде көрсетілген.



1.20-сурет – Қызмет диаграммасы

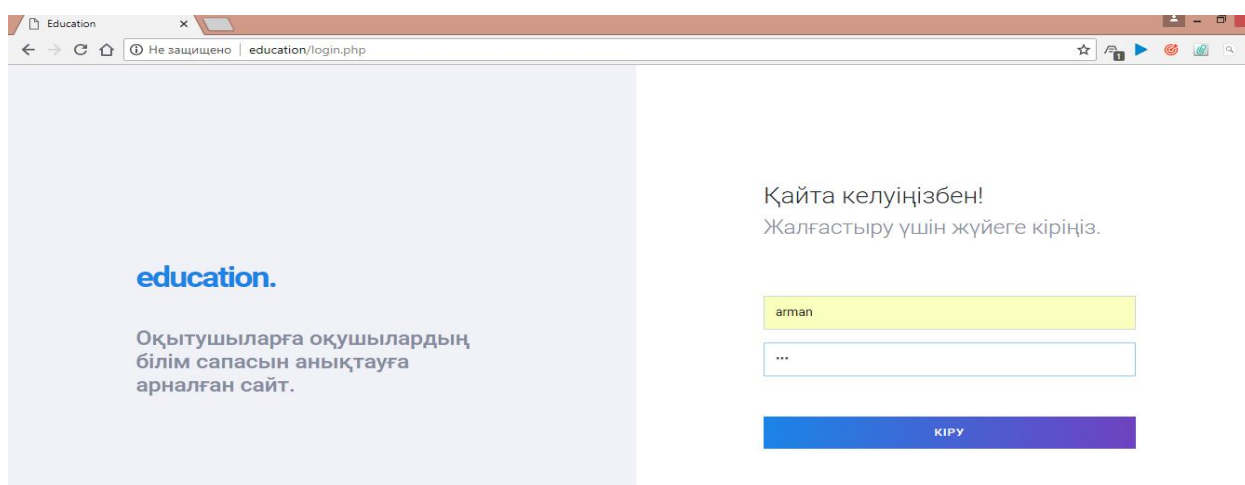
### 3.2 Программа құрылымы

Оқытушыларға оқушылардың білім сапасын анықтауға арналған веб-сайты барлық стандарттар, блоктар мен талаптарға сәйкес веб-беттерден тұрады. Осы бағытта сайт жасаудың таңдалған себебі қазіргі таңда сұранысқа ие болуы.

Сайтқа алдымен кіруден бұрын сол туралы түсіндіріп өтсем. Қазіргі таңда көптеген кеңселерде өздеріне арналған сайттары аз деп санауға болады. Ал бұл веб-сайттың осындай білім алушыларды дайындауға қажет. Себебі бұл сайттың басты мақсаты оқушылардың білім сапасын анықтау болып табылса, бұл веб-сайт арқылы білім берушілер өзінің күнделікті өткен ақпараттарын салып отыра алады және сол берген біліміне байланысты сол апта аралығында немесе өзі белгілеген уақыт бойынша білім алушылардың білім сапасын біліп отыра алады. Сол себепті ол белгілі бір уақыт аралығында тест салып отырады. Ал

білім алушы сол дайындалып жүрген сабағы бойынша, белгілі бір уақыт аралығында тапсыруына міндетті. Себебі ол оқушы төмен жинаған жағдайда мұғалімінің рейтингі түсіп кетуі ықтимал. Тіпті қазіргі таңда желілердің дамуына байланысты көпшілік оқушылар смартфонсыз жүрмейді. Кем дегенде бір смартфоннан болады. Олар тіпті қазір оқу құралдарын тасуды да азайтты. Сол себепті желінің дамуына байланысты олар үшін өткен тақырыптарға байланысты мәліметтерді жүктеп күнделікті тапсырманы орындап, сол алған ақпараттарына байланысты білім алушы тест тапсырады. Сол тапсырған тест арқылы ол өзінің білім сапасын баға арқылы біле алады. Ал бұл жердегі администратордың міндеті, сол мұғалімдер мен оқушыларды сайтқа тіркеп олар туралы ақпараттарын толтырып, оларға жеке кабинетіне кіруіне рұқсат береді(яғни логин және құпиясөз). Мұғалім мен оқушы өздері туралы ақпараттарын өзгертіп отыра қоймай олар құпиясөздерін де өзгерте алады. Яғни, ол веб-сайтқа кіру мақсатында қолданушы администратор, мұғалім немесе оқушы болуы тиіс. Бұл сайт белгілі бір кеңсе мүшелеріне арналған сайт деп айтсақ та болады. Осы сайт арқылы олар тек қана білім деңгейін біліп отыра қоймай, мұғалімдерінде рейтингтері шығып отырады. Мұғалімдер үшін электронды түрде ақпарат салып отырған ыңғайлы және күнделікті тапсырмаларын электронды түрде салып отырады. Веб-сайт оқушылар үшін де ыңғайлы. Олар электронды түрде оқып отыра алады. Сайт тек қана компьютерлік версиямен ғана емес, планшеттік және мобильдік версиямен кіріп отырса да болады. Егер де белгілі бір қателіктер шыққан жағдайда, олар администратормен байланысқа шығу арқылы немесе почтасына жазу арқылы шеше алады. Ал оқушылардың тапсырмаларына байланысты сұрақтары туындаса олар өз мұғалімдерінен сұрай алады.

Бұл сайтқа кіруге бізде 3 түрлі салада кіруіне мүмкіндік бар. Яғни, ол қолданушы администратор, мұғалім немесе оқушы болуы керек. Сол кезде ғана ол өзінің жеке кабинетіне кіруге мүмкіндігі бар. Ең алдымен бізде Сайтқа кірерден бұрын, education веб-сайтына кіруге сізден логин мен құпиясөзіңізді енгізіп кіруіңізді сұрайды. Жүйеге кіру беті 2.1-суретінде көрсетілген.



**2.1-сурет – Жүйеге кіру беті**

Келесі админ болып кірген кездегі админ бетінің көрсетілімі. Админ басты беті төмендегі 2.2-суретінде көрсетілген. Басты бетке кірген кезде профиль беті ашылады. Яғни, администратор өз профилін (2.5 және өзгерісі 2.6 суреттері) және құпиясөзін (өзгерісі 2.7 суреті) өзгерткен жағдайда ол қайта кіру бетінен кірген кезде ескі құпиясөзді енгізген жағдайда қателік шығарады(көрсетілімі 2.8 суреті). Профильді өзгерткен кезде ол өзінің жеке нөмері мен почтасын да өзгерте алады.

The screenshot displays the administrator's profile page. At the top left is the 'education.' logo. At the top right is a user profile icon labeled 'admin'. Below the logo is a navigation bar with five tabs: 'ПРОФИЛЬ' (selected), 'КУРСТАР', 'МУҒАЛІМДЕР', 'ОҚУШЫЛАР', and 'ШЫҒУ'. The main content area is titled 'ПРОФИЛЬ' and contains a profile card for 'Мейрамбек Арман Ролланұлы', Administrator. The card includes a profile picture, name, role, and two buttons: 'Профильді өзгерту' and 'Құпиясөз өзгерту'. Below the card are contact details: a phone number '8(707) 391-0679' and an email address 'MEIRAMBEK.ARMAN.97@GMAIL.COM'. To the right of the profile card is a statistics panel showing: 3 Courses, 1 Teachers, and 2 Students.

## 2.2-сурет – Администратор басты беті

Мұғалімдердің басты беті 2.3-суретінде көрсетілген.

The screenshot displays the teacher's profile page. At the top left is the 'education.' logo. At the top right is a user profile icon labeled 'test'. Below the logo is a navigation bar with three tabs: 'ПРОФИЛЬ' (selected), 'КУРСТАР', and 'ШЫҒУ'. The main content area is titled 'ПРОФИЛЬ' and contains a profile card for 'Шаяхметова Мейргуль', Teacher. The card includes a profile picture, name, role, and two buttons: 'Профильді өзгерту' and 'Құпиясөз өзгерту'. Below the card are contact details: a phone number '8(707) 777-7777' and an email address 'TEST@MAIL.RU'. To the right of the profile card is a statistics panel showing: 2 Courses and 3 Students.

## 2.3-сурет – Мұғалімдердің басты беті

Оқушылардың басты беті 2.4-суретінде көрсетілген.

The screenshot shows the 'education.' website interface. At the top left is the logo 'education.'. At the top right is a user profile icon labeled 'test\_stud'. Below the logo is a navigation bar with three items: 'ПРОФИЛЬ' (Profile) in a blue box, 'КУРСТАР' (Courses), and 'ШЫҒУ' (Logout). The main content area is titled 'ПРОФИЛЬ' and features a profile card for 'Жолшиева Кунсулу' (Zholsheva Kunsulu), a student. The card includes a circular profile picture, the name, and the role 'Оқушы' (Student). To the right of the profile card is a box displaying '3' and 'Курстар' (Courses). Below the profile card are two buttons: 'Профильді өзгерту' (Edit profile) and 'Күпиясөз өзгерту' (Change password). At the bottom of the profile card, there are two contact fields: a phone number '8(707) 777-7777' and an email address 'KUNSULU.01@GMAIL.KZ'.

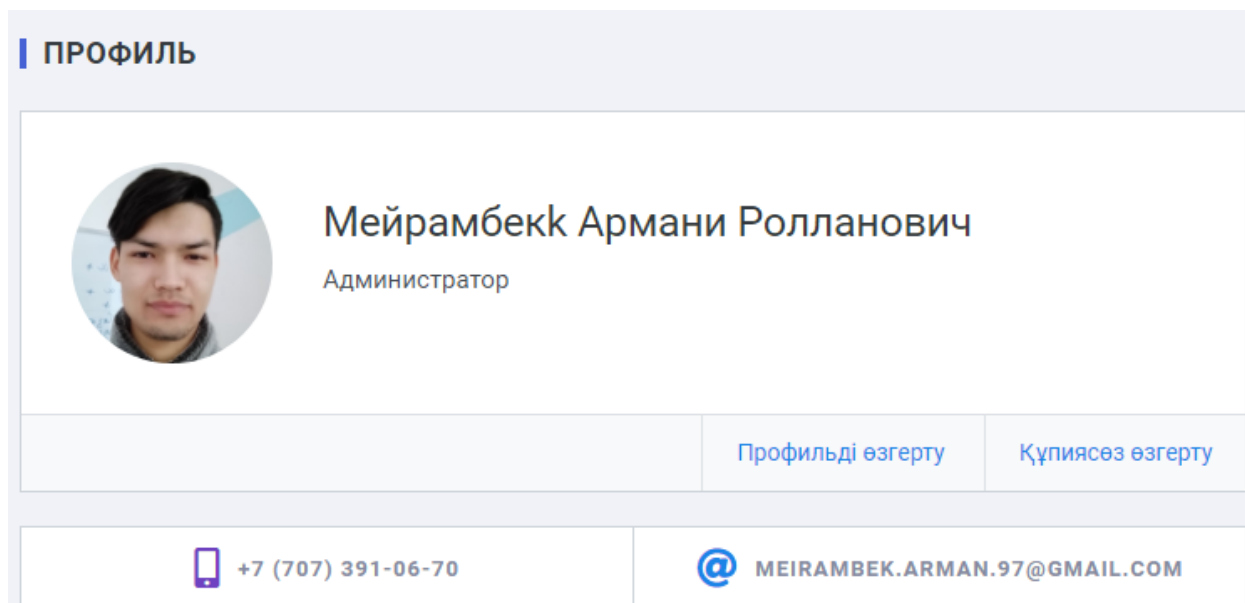
## 2.4-сурет – Оқушылардың басты беті

Профильді өзгерту беті 2.5-суретінде көрсетілген.

The screenshot shows the 'ПРОФИЛЬДІ ӨЗГЕРТУ' (Edit Profile) form. At the top is a header with the title and a close button 'x'. Below the header is a circular profile picture of a man. The form contains several input fields: 'Тері: \*' (Skin color) with the value 'Мейрамбекк' (Meirambek); 'Аты: \*' (Name) with the value 'Армани' (Armani); 'Әкесінің аты:' (Father's name) with the value 'Ролланович' (Rollanovich); a phone number field with '+7 (707) 391-06-70'; and an email field with 'meirambek.arman|97@gmail.com'. At the bottom right of the form are two buttons: 'Сақтау' (Save) and 'Жабу' (Close).

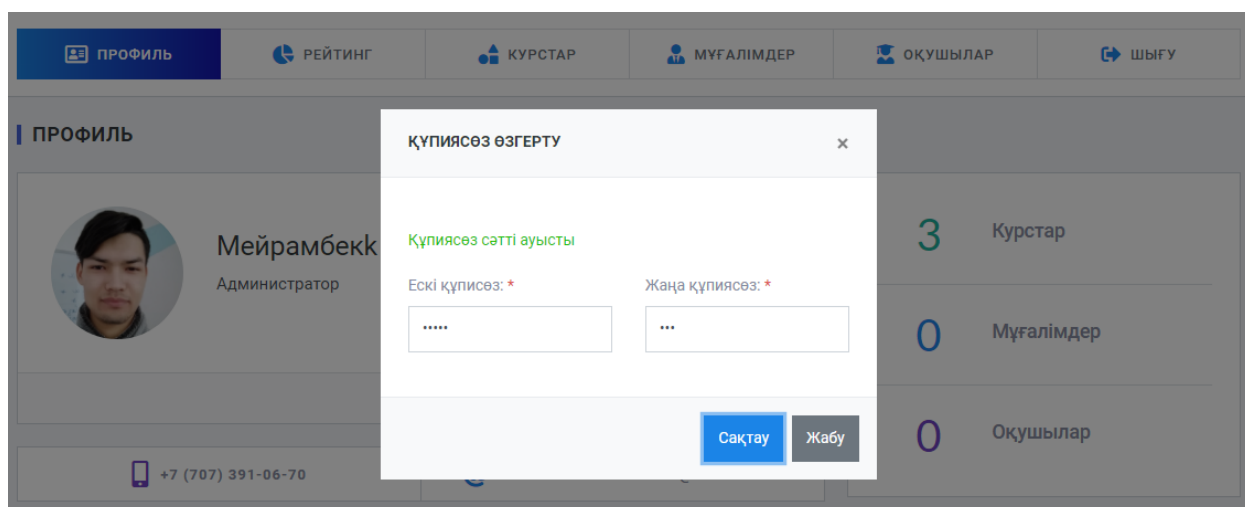
## 2.5-сурет – Профильді өзгерту беті

Профильдің өзгерілген кездегі нәтижесі 2.6-суретінде көрсетілген.



## 2.6-сурет – Өзгеріс сақталды

Құпиясөзді өзгерту 2.7-суретінде көрсетілген.

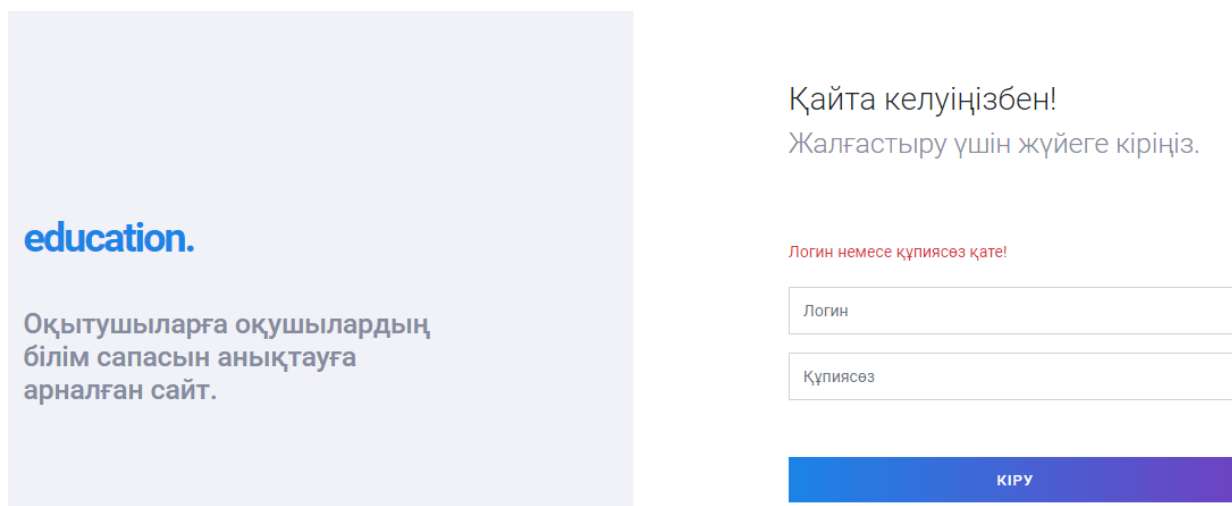


## 2.7-сурет – Құпиясөзді өзгерту

Құпиясөз дұрыс болмай қалған жағдайда бізге қателік шығарады. Яғни, біз құпиясөзді өзгерткен себептен ескі құпиясөзді дұрыс емес екендігін шығарып тұр. Бұл жерде құпиясөздің сәтті өзгергені көрініп тұр. Оның көрсетілімі 2.8-суретінде көрсетілген.



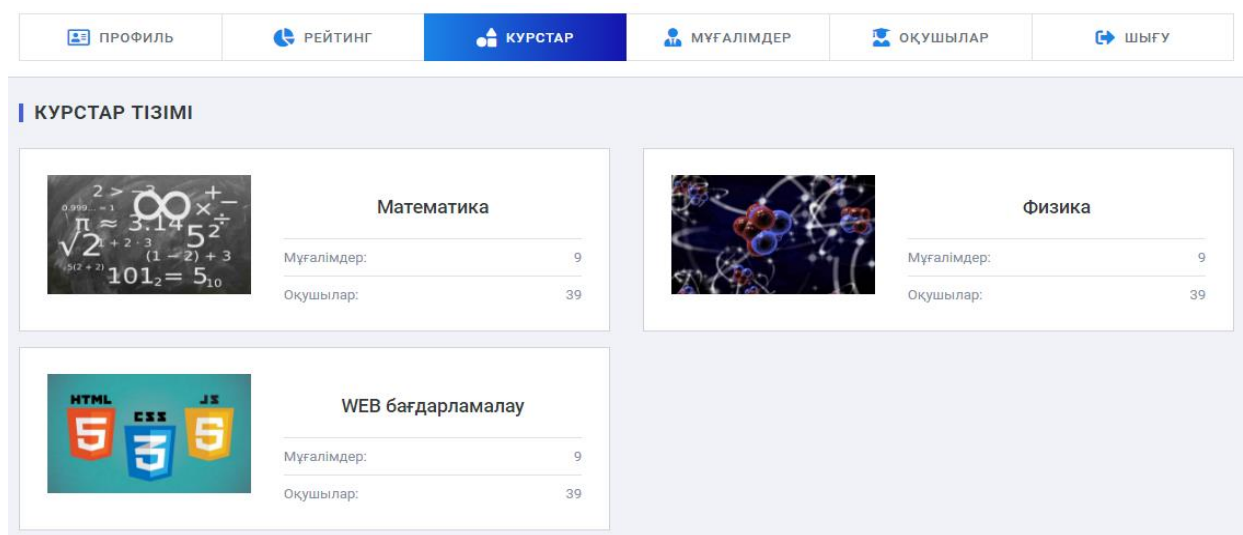
Құпиясөзді өзгертілгеннен кейінгі қателігі 2.8-суретінде көрсетілген.



The screenshot shows a login interface for an education website. On the left, there is a logo for 'education.' and a description: 'Оқытушыларға оқушылардың білім сапасын анықтауға арналған сайт.' On the right, there is a message: 'Қайта келуіңізбен! Жалғастыру үшін жүйеге кіріңіз.' Below this, a red error message reads: 'Логин немесе құпиясөз қате!'. There are two input fields: 'Логин' and 'Құпиясөз'. At the bottom, there is a blue button labeled 'КІРУ'.

## 2.8-сурет – Құпиясөз қателігі

Яғни, администратор мұғалімдер мен оқушылардың санын біліп оларға өзгерістер енгізе алады. Мұғалімдердің рейтингін көре алады. Мұғалім мен оқушының кірген кездегі профильдері де осы жоғарыдағы суреттерге қарап білуге болады. Тек олардың өзгешелігі мұғалім сабақтарына байланысты сабақтарды, тапсырмаларды және тесттарды салып отыра алады. Оқушыларының санын және жинаған баллдарын көре алады. Өзі таңдаған курстар тізіміне ғана қол жеткізеді және соларға өзгерістер енгізе алады. Жоғарыда көрсетіп өткендей өз профилін өзгерте алады. Ал оқушылардың өзгешелігі дәл мұғалімдікіне ұқсас, бірақ олар тікелей сол таңдаған пәніне байланысты сабақтарды, тапсырмаларды жүктей алады және соған орай тестті тапсырып өз бағаларын көре алады. Оқушылар да өз профильдеріне өзгерістер енгізе алады. Курстар бетін төмендегі 2.9-суретінде көрсетілген.



The screenshot shows a navigation menu at the top with options: 'ПРОФИЛЬ', 'РЕЙТИНГ', 'КУРСТАР' (highlighted), 'МҰҒАЛІМДЕР', 'ОҚУШЫЛАР', and 'ШЫҒУ'. Below the menu, the 'КУРСТАР ТІЗІМІ' section displays three course cards:

Course Name	Teachers (Мұғалімдер)	Students (Оқушылар)
Математика	9	39
Физика	9	39
WEB бағдарламалау	9	39

## 2.9-сурет – Курстар тізімі

## ҚОРЫТЫНДЫ

Дипломдық жұмысты орындау барысында қолданыстағы сайттарға шолу жасалды және қажетті ақпараттар таңдалды. Web-қосымшаны жасау құралдарына талдау жасалды. Сайттың құрылымы және графикалық интерфейсін жасап, web-қосымша беттерінің арасында байланыс жасалды.

Web-қосымша қолданушыға түсінікті қарапайым және ыңғайлы навигациялық жүйеде қарастырылды.

Бұл дипломдық жұмысты орындау барысындағы негізгі мақсат – білім жүйесін құру. Білім алушының білімін тез арада толықтырып және тексеру. Education (электронды оқу) жүйесі өте жиі қолданылып жатыр, өйткені Education технологиясы білім алушының дайындық деңгейі мен дәрежесін жылдам және сенімді тексеру әдісі болып табылады.

Жүйені құру үшін Apache HTTP сервері, MySQL мәліметтер базасын басқару жүйесі, Open Server, JavaScript(jQuery және Ajax) программалау тілі, CSS стилдерді сипаттау тілі, Bootstrap – құралдар жиынтығы, Grid система, PHP скрипті бағдарламалау тілдері, GitHub және phpMyAdmin – веб-қосымшасы қолданылды.

## ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Баялы Ә.Т. Web бағдарламалау негіздері. -Қ.А.Ясауи атындағы Халықаралық қазақ-түрік университетінің "Тұран" баспасы, 2013. - 124 б.
- 2 М. В. Кузнецов, И. В. Симидянов - PHP 5/6.- СПб. : БХВ-Петербург, 2010. - 2014 с.
- 3 Ахметова О. С. Основы компьютерных технологий. - Алматы : КазНПУ им. Абая, 2006. - 263 с.
- 4 Мухаметшин Д.Ф. Описываются приёмы работы с элементами HTML-программирования PHP. – СПб.: БХВ – Питер, 2000. – 120 с.
- 5 Бөрібаев, Б. Программалау технологиялары. - Алматы : ЖШС РПБК "Дәуір", 2011. - 352 б.
- 6 Максимов С.М. MySQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: BHV, 2004. – 416с.
- 7 Дронов, В. А. HTML 5, CSS 3 и Web 2.0 разработка современных Web- сайтов. - СПб. : БХВ-Петербург, 2011. - 416 с.
- 8 Шафер С. HTML, XHTML и CSS. Библия пользователя. 5 .изд. - М.: Диалектика, 2010. - 656 с.
- 9 Макфарланд Д.С. JavaScript и jQuery. Исчерпывающее руководство 2015. – 143 с.
- 10 Дэвид Скляр. Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов, 2017. – 150 с.
- 11 Вагнер Р Энциклопедия HTML. – Киев: Bhv, 2001. – 400 с.
- 12 Веймаер, Р.; Сотел, Р. Освой самостоятельно Microsoft SQL Server 2000 за 21 день (+ CD-ROM); М.: Вильямс, 2013. -549 с.
- 13 Дэвидсон, Луис проектирование баз данных на SQL Server 2000; Бином, 2009. -631 с.
- 14 Шнайдер, Роберт Microsoft SQL Server 6.5. Проектирование высокопроизводительных баз данных; М.: Лори, 2010. - 361 с.
- 15 Максим К., Игров С., Сергей Г. PHP 5: Практика создание Web – сайтов. СПб.: БХВ – Петербург, 2007. – 150 с.
- 16 Лаура Томсон, Люк Веллинг. Разработка Web-приложений на PHP и MySQL. – СПб.: БХВ – Петербург, 2008. – 365 с.

## **А қосымшасы (міндетті)**

### Техникалық тапсырма

#### **А.1 Кіріспе**

Қазіргі таңда веб-сайт құрылғылардың жаңаша нұсқаларын көре отырып, заманның жетілгенің белгісін байқай аламыз. Веб-сайт катынасу құрылғыларды жоғары деңгейде жетіліп, дамуын бір ұғыммен анықтап айтуға болады, ол – технологиялардың біріктірілуі.

«Education» білім орталығы – бұл пайдаланушыларға ыңғайлы түрде білім алуларына және керекті ақпараттарын қайталап естеріне түсірулеріне мүмкіндік береді. Дипломдық жобаның басты мақсаты – Education білім орталығының веб-сайтын құру негізделген.

##### **А.1.1 Өңдеудің мақсаты мен қызметі**

Басты талап, пайдаланушы ақпараттарды сақтау немесе жүктеу мүмкіндігіне ие болады. Соған қоса олар сол жүктелген ақпараттар бойынша қайталау арқылы тест тапсыру мүмдігіне ие. Әсіресе бұл көптеген қолданушыларға уақытты тиімді пайдалануға көмегін тигізеді.

Пайдаланушы ең бастысы веб-сайтқа администратор арқылы тіркелсе болды. Пайдаланушы кез-келген уақытта, кез-келген жерде, қажетті сәтте бұл қосымшаны пайдалана алады.

##### **А.1.2 Қолдану саласы**

Қолдану аймағы – IT саласындағы мамандар арасында.

##### **А.1.3 Анықтамалар, терминдер және қысқартулар**

«Education» білім орталығы – бұл білім беру жүйесі.

Open Server – көп функционалды басқару бағдарламасы және қосылған компоненттердің үлкен таңдауымен тасымалданатын жергілікті WAMP / WNMP сервері.

## **А қосымшасының жалғасы**

UML – бағдарламалық қамтаманың «сызбасын» көрсететін стандартты құрал.

Модельдеу тілі – бұл жүйенің тұжырымдамалық және физикалық көрінісіне бағытталған ереже, тіл және сөздік.

GitHub – жоба мен код нұсқасын басқару жүйесі.

CASE – технологиясының тұжырымдамасы компьютерді қолдана отырып, компьютерлік бағдарламалық қамтамасыз ету / жүйелік инженерия – дамушы жүйелер болып табылады.

### **А.2 Функциональды диаграммалар**

Rational Software-бағдарламалық қамтамасыз етуді әзірлеуші компания. 2003 жылға дейін Rational тәуелсіз компания болды, 2003 жылы оны IBM корпорациясы іске асырды. Компанияның көптеген өнімдері модельдеу үшін, сондай-ақ бағдарламалық жасақтаманы әзірлеу және қолдау үшін арналған.

Компанияда Rational Unified Process (RUP) бағдарламалық қамтамасыз етуді әзірлеу әдістемесі әзірленді. Әдіснамада әзірлеудің барлық кезеңдері бойынша ұсынымдар беріледі: бизнесті үлгілеуден бастап тестілеуге дейін және дайын бағдарламаны пайдалануға тапсырғанға дейін.

Жобамен жұмыс істеудің жалпы процесі диаграммаларға сәйкес графигтік элементтерді қосумен, осы элементтер арасында қатынастар орнатумен, оларды сипаттаумен және құжаттаумен шектеледі.

## В қосымшасы (міндетті)

### Бағдарламалау мәтіні

#### *1 Index.php бөлімі*

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Education</title>
    <link href="lib/font-awesome/css/font-awesome.css" rel="stylesheet">
    <link href="lib/Ionicons/css/ionicons.css" rel="stylesheet">
    <link href="lib/medium-editor/css/medium-editor.css" rel="stylesheet">
    <link href="lib/medium-editor/css/default.css" rel="stylesheet">
    <link rel="stylesheet" href="css/slim.css">
    <link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.8.1/css/all.css" integrity="sha384-
50oBUHEmvpQ+11W4y57PTFmhCaXp0ML5d60M1M7uH2+nqUivzIebhndOJK28a
nvf" crossorigin="anonymous">
  </head>
  <body>
    <?php
      session_start();
      require("config.php");
      if(isset($_SESSION['login_education'])){
        $query_user_info = mysql_query("SELECT * FROM db_users u,
db_users_type t
                                WHERE u.login = '$_SESSION[login_education]'
                                AND t.user_type = u.user_type ");
        $user_info = mysql_fetch_array($query_user_info);
        require "pages/modules/header.php";
        require "pages/modules/main.php";
        require "pages/content.php";
      }else header('Location:login.php');
    ?>
    <script src="lib/jquery/js/jquery.js"></script>
    <script src="lib/popper.js/js/popper.js"></script>
    <script src="lib/bootstrap/js/bootstrap.js"></script>
    <script src="lib/jquery.cookie/js/jquery.cookie.js"></script>
    <script src="lib/medium-editor/js/medium-editor.js"></script>
```

## В қосымшасының жалғасы

```
<script src="lib/jquery.maskedinput/js/jquery.maskedinput.js"></script>
<script src="js/script.js"></script>
<script src="js/ajax.js"></script>
</body>
</html>
```

### 2 Config.php бөлімі

```
<?php
    $user = 'root';
    $pass = "";
    $data = 'education';
    $db = mysql_connect('localhost', $user, $pass, $data);
    mysql_select_db($data, $db);
    mysql_set_charset("utf8");
?>
```

### 3 Header.php бөлімі

```
<div class="slim-header" style="margin-top: -20px;">
    <div class="container">
        <div class="slim-header-left">
            <h2 class="slim-logo"><a href=".">education<span>.</span></a></h2>
        </div>
        <div class="slim-header-right">
            <div class="dropdown dropdown-c">
                <a href="#" class="logged-user" data-toggle="dropdown">
                    
                    <span> <?=$_SESSION['login_education']?> </span>
                </a>
            </div>
        </div>
    </div>
</div>
```

### 4 Header.php бөлімі



## В қосымшасының жалғасы

```
<div class="slim-navbar" style="margin-top: -21px;">
  <div class="container">
    <ul class="nav">
      <?php
        if($_SESSION['user_type'] == '1'){
          if(isset($_GET['profile'])) echo '<li class="nav-item active"><a
class="nav-link" href="?profile"> <i class="icon far fa-address-card"></i>
<span>Профиль</span> </a></li>';
          else echo '<li class="nav-item"><a class="nav-link" href="?profile"> <i
class="icon far fa-address-card"></i> <span>Профиль</span> </a></li>';
          if(isset($_GET['objects'])) echo '<li class="nav-item active"> <a
class="nav-link" href="?objects"> <i class="icon fas fa-shapes"></i>
<span>Курстар</span> </a> </li>';
          else echo '<li class="nav-item"> <a class="nav-link" href="?objects"> <i
class="icon fas fa-shapes"></i> <span>Курстар</span> </a> </li>';
          if(isset($_GET['tutors'])) echo '<li class="nav-item active"> <a
class="nav-link" href="?tutors"> <i class="icon fas fa-user-tie"></i>
<span>Мұғалімдер</span> </a> </li>';
          else echo '<li class="nav-item"> <a class="nav-link" href="?tutors"> <i
class="icon fas fa-user-tie"></i> <span>Мұғалімдер</span> </a> </li>';
          if(isset($_GET['students'])) echo '<li class="nav-item active"> <a
class="nav-link" href="?students"> <i class="icon fas fa-user-graduate"></i>
<span>Оқушылар</span> </a> </li>';
          else echo '<li class="nav-item"> <a class="nav-link" href="?students"> <i
class="icon fas fa-user-graduate"></i> <span>Оқушылар</span> </a> </li>';
        }else if($_SESSION['user_type'] == '2'){
          if(isset($_GET['profile'])) echo '<li class="nav-item active"><a
class="nav-link" href="?profile"> <i class="icon far fa-address-card"></i>
<span>Профиль</span> </a></li>';
          else echo '<li class="nav-item"><a class="nav-link" href="?profile"> <i
class="icon far fa-address-card"></i> <span>Профиль</span> </a></li>';
          if(isset($_GET['objects'])) echo '<li class="nav-item active"> <a
class="nav-link" href="?objects"> <i class="icon fas fa-shapes"></i>
<span>Курстар</span> </a> </li>';
          else echo '<li class="nav-item"> <a class="nav-link" href="?objects"> <i
class="icon fas fa-shapes"></i> <span>Курстар</span> </a> </li>';
        }else if($_SESSION['user_type'] == '3'){
          if(isset($_GET['profile'])) echo '<li class="nav-item active"><a
class="nav-link" href="?profile"> <i class="icon far fa-address-card"></i>
<span>Профиль</span> </a></li>';
```

## В қосымшасының жалғасы

```
else echo '<li class="nav-item"><a class="nav-link" href="?profile"> <i
class="icon far fa-address-card"></i> <span>Профиль</span> </a></li>';

if(isset($_GET['objects'])) echo '<li class="nav-item active"> <a
class="nav-link" href="?objects"> <i class="icon fas fa-shapes"></i>
<span>Курстап</span> </a> </li>';
else echo '<li class="nav-item"> <a class="nav-link" href="?objects"> <i
class="icon fas fa-shapes"></i> <span>Курстап</span> </a> </li>;}
?>
<li class="nav-item"><a class="nav-link" href="./logout.php"> <i
class="icon fas fa-sign-out-alt"></i> <span>ШЫҒУ</span> </a></li>
</ul>
</div>
</div>
```

### 5 Content.php бөмі

```
<?php
if($_SESSION['user_type'] == '1'){
    if(isset($_GET['profile'])) require "pages/admin/profile.php";
    else if(isset($_GET['objects'])) require "pages/admin/objects.php";
    else if(isset($_GET['tutors'])) require "pages/admin/tutors.php";
    else if(isset($_GET['students'])) require "pages/admin/students.php";
    else header('Location:./?profile');
}else if($_SESSION['user_type'] == '2'){
    if(isset($_GET['profile'])) require "pages/tutor/profile.php";
    else if(isset($_GET['objects'])) require "pages/tutor/objects.php";
    else header('Location:./?profile');
}else if($_SESSION['user_type'] == '3'){
    if(isset($_GET['profile'])) require "pages/student/profile.php";
    else if(isset($_GET['objects'])) require "pages/student/objects.php";
    else header('Location:./?profile');}
?>
```

### 6 Login.php бөмі

```
<!DOCTYPE html>
```

## В қосымшасының жалғасы

```
<html>
<head>
  <meta charset="utf-8">
  <title>Education</title>
  <link href="lib/font-awesome/css/font-awesome.css" rel="stylesheet">
  <link href="lib/Ionicons/css/ionicons.css" rel="stylesheet">
  <link rel="stylesheet" href="css/slim.css">
</head>
<body>
  <div class="d-md-flex flex-row-reverse">
    <div class="signin-right">
      <div class="signin-box">
        <h2 class="signin-title-primary">Қайта келуіңізбен!</h2>
        <h3 class="signin-title-secondary">Жалғастыру үшін жүйеге
кіріңіз.</h3>
        <?php
session_start();
require("config.php");
if(isset($_POST['log_in'])){
  $login = $_POST['login'];
  $pass = $_POST['password'];
  $login = stripslashes($login);
  $login = htmlspecialchars($login);
  $login = trim($login);
  $pass = stripslashes($pass);
  $pass = htmlspecialchars($pass);
  $pass = trim($pass);
  $password = md5($pass);
  $user = mysql_query("SELECT * FROM db_users
                        WHERE login = '$login'
                        AND password = '$password' ");
  $row = mysql_fetch_array($user);
  if($row['user_id'] != ""){
    $_SESSION['login_education'] = $row['login'];
    $_SESSION['user_type'] = $row['user_type'];
    header("Location: ./");}
  else{
    echo "<p class='text-danger'> Логин немесе құпиясөз қате! </p>";}}
?>
<form action="login.php" method="post" required>
  <div class="form-group">
```

## В қосымшасының жалғасы

```
        <input type="text" name="login" class="form-control"
placeholder="Логин" required>
    </div>
    <div class="form-group mg-b-50">
        <input type="password" name="password" class="form-control"
placeholder="Құпиясөз" required>
    </div>
    <input type="submit" name="log_in" class="btn btn-primary btn-block
btn-signin" value="Кіру">
    </form>
</div>
</div>
<div class="signin-left">
    <div class="signin-box">
        <h2 class="slim-logo"><a
href="index.html">education<span>.</span></a></h2>
        <h3>Оқытушыларға оқушылардың білім сапасын анықтауға
арналған сайт.</h3>
    </div>
</div>
</div>
<script src="lib/jquery/js/jquery.js"></script>
<script src="lib/popper.js/js/popper.js"></script>
<script src="lib/bootstrap/js/bootstrap.js"></script>
<script src="js/script.js"></script>
</body>
</html>
```

### 7 Logout.php бөлімі

```
<?php
session_start();
session_unset();
header('Location:./');
?>
```



