

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

Файзулла Ә.

«Find System» проектісінің веб бетін және REST сервисін құру

Дипломдық жобаға  
**ТҮСІНІКТЕМЕЛІК ЖАЗБА**

5B060200 – «Информатика» мамандығы

Алматы 2019

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

«17» май 2019 ж.

Дипломдық жобаға  
**ТҮСІНІКТЕМЕЛІК ЖАЗБА**

«Find System» проектісінің веб бетін құру

5B060200 – «Информатика» мамандығы

Орындаған

Файзулла Ә.

Ғылыми жетекші

Сениор-лектор

С. Қалдыбеков

«17» мамыр 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

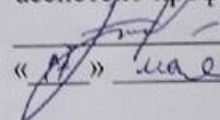
Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

5B060200 – «Информатика»

**БЕКІТЕМІН**

ПИ кафедра меңгерушісі,  
техника ғылымдарының кандидаты,  
ассистент-профессор

 Р. Юнусов  
«17» мае 2019 ж.

**Дипломдық жоба орындауға  
ТАПСЫРМА**

Білім алушы Файзулла Әйгерім

Тақырыбы «Find System» проектісінің веб бетін құру

Академиялық мәселелер жөніндегі проректорының 2018 жылғы «16» қазан № 1162-б бұйрығымен бекітілген

Аяқталған жобаны тапсыру мерзімі 2019 жылғы «20» мамыр

Дипломдық жобаның бастапқы берілістері Ұсынылатын дипломдық жобада «Find System» проектісінің веб бетін құру. Веб-бет құру барысында PHP бағдарламалу тілінің Yii 2.0 тірегін қолданамын.

Дипломдық жобада қарастырылатын мәселелер тізімі:

а) Негізгі бөлім;

б) Жобалау бөлімі;

е) Қолданылған бағдарламалық қамтамалар тізімі;

ж) Веб бетті құру және оны іске асыру.

Сызба материалдарының тізімі (міндетті сызбалар дәл көрсетілуі тиіс)

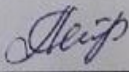
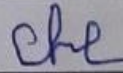
Жобаның презентациялық 25 слайды ұсынылған.

Ұсынылатын негізгі әдебиет 16 әдебиеттер тізімінен.

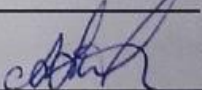
Дипломдық жобаны дайындау  
КЕСТЕСІ

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

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	16.05.2019	
Бағдарламалық бөлімі	Сман Н.О. Ассистент	16.05.2019	

Ғылыми жетекші  С. Қалдыбеков

Тапсырманы орындауға алған білім алушы  Ә. Файзулла

Күні «\_\_» \_\_\_\_\_ 2019 ж.



## АҢДАТПА

Қазіргі кезде барлық адамдар мобильді телефонды өте көп қолданады. Бүгін, адамдардың өмірінде ұялы телефон рөлін асыра бағалау қиын. Кез келген уақытта осы ықшам құрылғымен бізді қызықтыратын ақпаратты табуға болады. Барлығы ноутбук, ұялы телефонды әрдайым қолданады. Осы себепті жобаның клиенттік бөлімі Android платформасында жасалынған.

Бұл дипломдық жұмыстың мақсаты REST технологиясын пайдалана отырып университеттің оқу жүйесін автоматтандыру сонымен қатар университетімізде оқитын студенттерге, оқытушыларға немесе абитуриенттерге аудиторияның орнын табуға және аудитория жайлы ақпарат алуды жеңілдету және жылдамдату. Брондау жүйесі оқытушыларға қалаған аудиториясында сабақ беру мүмкіндігін береді.

Сайт үлкен аудиторияға арналған (кез-келген адамға), сондықтан сайт жоғарғы технологияларға сәйкес болуы керек, тартымды дизайнға, және ыңғайлы интерфейске ие болуы керек, және де, көпфункционалды, кез-келген веб-қолданушыға ыңғайлы болуы керек.

Жоба бойынша, Android мобильді қосымшасы клиент ретінде, ал веб-бет сервер ретінде, клиент-сервер технологиясын іске асырады.

REST сервис тек қана Android платформасына ғана емес, кез-келген құрылғыға, яғни Android, IOS, web-бет, Windows операциялық жүйелерінің барлығына веб-бетті қолдануға мүмкіндік береді.

## АННОТАЦИЯ

В настоящее время практически все люди пользуются мобильным телефоном. Сегодня трудно переоценить роль мобильного телефона в жизни людей. В любое время с этим компактным устройством можно найти интересующую нас информацию. Все всегда используют ноутбук, мобильный телефон. По этой причине клиентская часть проекта разработана на платформе Android.

Целью данной дипломной работы является автоматизация учебной системы университета с использованием технологии REST, а также облегчение и ускорение получения информации о аудитории и местоположении студентов, преподавателей или абитуриентов, обучающихся в университете. Система бронирования дает возможность преподавателям преподавать в желаемой аудитории.

Сайт предназначен для большой аудитории( для любого человека), поэтому сайт должен соответствовать высоким технологиям, обладать привлекательным дизайном, и удобным интерфейсом, а также многофункциональным, удобным для любого веб-пользователя.

По проекту, мобильное приложение Android, как клиент, а веб-страница, как сервер, реализует технологию клиент-сервер.

REST сервис позволяет использовать веб-страницу не только на платформе Android, но и на любом устройстве, включая все операционные системы Android, IOS, web-страница, Windows.

## ANNOTATION

Nowadays all people use mobile phone very much. Today it is difficult to overestimate the role of a mobile phone in people's lives. At any time with this compact device you can find the information we need. Everyone always uses a laptop, a mobile phone. For this reason, the client part of the project is developed on the Android platform.

The purpose of this thesis is to automate the University's educational system using REST technology, as well as facilitate and accelerate the receipt of information about the audience and the location of students, teachers or students studying at the University. The booking system allows teachers to teach in the desired audience.

The site is intended for a large audience( for any person), so the site must comply with high technology, have an attractive design and user-friendly interface, as well as multifunctional, convenient for any web user.

According to the project, the Android mobile application as a client and the web page as a server implements the client-server technology.

REST service allows you to use a web page not only on the Android platform, but also on any device, including all operating systems Android, IOS, web-page, Windows.

## МАЗМҰНЫ

	КІРІСПЕ	8
1	Негізгі бөлім	9
1.1	Жобаның өзектілігі	9
1.2	Веб-қосымшалардың сервис-бағытталған архитектурасы	9
1.2.1	SOAP Веб-қызметі	10
1.2.2	REST сервистер құрылымы	11
1.3	Ұқсас жобаларға шолу	14
2	Жобалау бөлімі	15
2.1	UML тілінде модельдеу	15
2.2	Rational rose бағдарламалық қамтамасы	15
2.3	Прецеденттер диаграммасы	16
2.4	Тізбек диаграммасы	17
2.5	Күй диаграммасы	17
2.6	Класстар диаграммасы	18
2.7	Компонент диаграммасы	19
2.8	ER диаграмма құру	19
3	Қолданылған бағдарламалық қамтамалар	20
3.1	MySQL технологиясы	20
3.2	JavaScript бағдарламалау тілі	22
3.3	HTML тілі туралы түсінік	24
3.4	Apache HTTP сервері	25
3.5	PHP программалау тілі	26
3.6	YII2.0 тірегі	28
4	Веб бетті құру және іске асыру	30
4.1	«Find System» проектісі туралы	30
4.2	Қолданушының интерфейсі	30
	Қорытынды	38
	Пайдаланылған әдебиеттер тізімі	39
	А қосымшасы	40
	Б қосымшасы	44



## КІРІСПЕ

Интернет бүгінгі таңда ақпараттың кең ауқымын қамтитын ақпараттың кезі болып табылады. Интернет ақпараттық кеңістікті құрайды. Сол кеңістіктегі құжаттарды web беттер деп атайды.

Қазіргі уақытта сол веб-ресурстардың өсу қарқыны артып келеді. Көптеген адамдар үшін компьютерден алшақтамай ұйымдардың қызметтерімен олардың веб – сайттары арқылы танысу өте ыңғайлы. Соның бір дәлелі осы дипломдық жоба болып табылады.

Дипломдық проектінің өзектілігі бұл веб сайттың көп функционалды болып табылуы. Проектінің негізгі мақсаты REST технологиясын пайдалана отырып, веб сайт жасап шығу. Бұл веб сайт үлкен аудиторияға арналғандықтан, заманауи технологияларға сәйкес болуы және қолданушыға ыңғайлы болуы қажет.

Университетімізге арналған “Find system” веб-бетін құру студенттерге, оқытушыларға кез келген жерде және кез келген уақытта университетіміз туралы барлық қажетті ақпаратты алуға мүмкіндік береді. Проектінің түпкі мақсаты, университетімізде оқитын студенттерге, оқытушыларға немесе абитуриенттерге аудиторияның орнын табуға және аудитория жайлы ақпарат алуды жеңілдету және жылдамдату. Брондау жүйесі оқытушыларға қалаған аудиториясында сабақ беру мүмкіндігін береді.

- “Find system” проектісінде қолданылатын технологиялар:
- PHP программалау тілі,
- MySQL деректер қоры,
- Apache веб-сервері,
- HTML,
- CSS,
- Yii 2.0 PHP тілінде жазылған тірек,
- Bootstrap.

Бұл жоба веб парақтан және Android мобильді қосымшасынан тұрады. Веб – бет пен мобильді қосымша арасындағы байланыс RESTful API көмегімен жүзеге асады.

## **1 Негізгі бөлім**

### **1.1 Жобаның өзектілігі**

«Find System» проектісінде бұл веб бет клиент–сервер технологиясындағы сервердің рөлін атқарса, ал мобильді қосымша клиент болып табылады. Бұл жеке веб бет болып табылады. Веб бет үлкен аудиторияға арналған. Сондықтан әрбір кез келген адам, университетіміздің оқытушылары мен студенттері және оқуға жаңа түскен абитуриенттер бұл веб сайтты қолдана алады. Жобаның өзектілігі REST технологиясының мүмкіндіктерін кеңінен пайдаланып, мобильдік қосымшамен байланыс жасай алуында. Бұл жоба Php тілінде, Yii2.0 фреймворгін қолдана отырып жазылды. Бұл тірек қазіргі кезде қарқынды дамуда, сондықтан жобаны басқаруға қолайлы.

Әрбір заманауи адам технологиядағы соңғы жаңалықтарды қадағалауға тырысады. Ұялы телефон өз міндеттерін ғана емес, басқа да көптеген қызметтерді атқарады. Олардың жүзеге асырылуын жеңілдету үшін әзірлеушілер мобильді қосымшаның түрлі түрлерін шығаруда. Олар жұмыс ыңғайлылығына үлес қосады және пайдаланушының уақытын үнемдеуге көмектеседі. Қазіргі әлемде ұялы қосымшаны әзірлеу өте танымал. Мобильді қосымшада бірнеше артықшылықтар бар. Олар интерактивті, көпфункционалды және сенсорлық. Мобильді қосымшалардың көмегімен желіден тыс жұмыс істеуге болады. Қазіргі уақытта мобильді қосымшалар компьютер бола алмайды, бірақ оның мүмкіндіктерін орындағаннан кем емес. «Find System» проектісінің негізгі мақсаты студенттер мен оқытушыларға аудиторияны іздеуді қолайландыру болып табылады. Жоба веб сайт пен мобильді қосымшадан тұрады. Ұялы қосымша деректерді веб сайттан алады.

### **1.2 Веб-қосымшалардың сервис-бағытталған архитектурасы**

SOA – полиморфизм мен инкапсуляцияға ұқсайтын, басқа технологиялар мен өнімдерге тәуелді емес архитектуралық қағидалардың жиынтығы. Веб сервистерді технологияға тәуелсіз жариялауға, қолдануға және стандартты түрде пайдалануға болады. SOA веб сервисі жалпы мақсаттағы API емес, ол HTTP арқылы деректер қорына қолжетімділік беретін GRUD сервистер. SOA бұл тек қана сервистік архитектура емес, сонымен бірге бұл әдістемелер, саясаттар және тіректердің жиынтығы. Сол арқылы біз қажетті сервисті ұсынамыз және аламыз.

**Артықшылықтары**

– технологиялардың жиынтығының және сервистерді масштабтаудың тәуелсіз болуы.

– стандартты және қауіпсіз желі каналы (HTTP арқылы тексті 80 порт пен жіберу)

- хабарлама алмасудың оңтайландырылған жолы
- тұрақты хабарлама алмасудың ерекшеліктері
- домен контекстінің оқшаулануы

Кемшілігі

– әр түрлі веб - қызметтер хабарлама жіберудегі тілдердің өзгешелігіне байланысты интеграциялауы қиынға соғады. Мысалы әр түрлі JSON форматты қолданатын екі веб - қызмет.

- синхронды түрде хабар алмасу жүйені қайта жүктеуі мүмкін.

Клиент пен SOAP және REST сервисі арасындағы коммуникацияны жүзеге асыратын екі сервисі қарастырып көрейік.

### **1.2.1 SOAP Веб-қызметі**

SOAP – бұл машиналық өңдеу тұрғысынан қиын деп есептелінетін күрделі хаттамалар мен стандарттардың толық жиынтығы. Сондықтан REST SOAP қа қарағанда тез жұмыс істейді.

SOAP көліктік хаттама ретінде HTTP ді қолданады, ал REST технологиясы осы HTTP ға негізделген. Бұл сервер деңгейінде кәширлеудің барлық жұмысы, масштабтау және HTTP негізінде REST архитектурада жалғасуда, және сіз SOAP үшін басқа құралдарды іздеуіңіз керек. REST бұл бірнеше форматта беріле алады, ал SOAP тек XML форматымен жұмыс жасайды. Негінде REST ті қарапайымдылық деп, SOAP ты стандарттар деп айтуға болады. Веб қызметтерді толық сипаттау үшін біз SOAP технологиясында WSDL хаттамаларын қолданамыз.

SOAP толықтай стандартталған. SOAP негізінде операциялармен жұмыс жасайды, ал REST ресурстармен. SOAP пен REST салыстырудың қажеті жоқ, өйткені біріншісі бұл протокол, ал екіншісі архитектуралық. SOAP пен REST негізгі артықшылығы клиент-сервер арасындағы қатынастың өзара әрекеттесу дәрежесі десек болады. SOAP қолданушысы сервермен тығыз байланысқан, үстел үстіндегі қолданушы қосымшаасы ретінде жұмыс жасайда. Ол жерде клиент-сервер байланысы өте жақсы, егер біреуінде бір ақаулық кететін болса бүкіл система жұмыс жасамайды.

SOAP бұл объектілерге қарапайым қатынас протоколы. SOAP қысқа мерзімді транзакциялар және ұзақ мерзімді транзакциялар үшін транзакцияларды басқару үшін ACID негізіндегі транзакцияларға кешенді қолдау көрсетуге ие. Ол сондай-ақ бөлінген ресурстар арқылы екі фазалық берілісті қолдайды.

SOAP бұл хаттама, сондықтан REST ті пайдалана алмайды. REST технологиясы SOAP тың веб – қызметін пайдалана алады, өйткені HTTP және SOAP сияқты кез келген концепцияларды пайдалана алады.

## 1.2.2 REST сервистер құрылымы

REST (RESTful) – HTTP протоколы арқылы қосымшамен/сайтпен өзара іс-қимылды ұйымдастырудың жалпы принциптері болып табылады. Әрбір сұрау пайдаланушыға ақпаратты (мысалы, авторландыру, OAuth арқылы алынған token) және операция үшін қажетті барлық параметрлерді жібереді.

REST жүйелік ресурстарға бағдарланған веб-сервистерді жобалау үшін архитектуралық қағидалар жиынтығын анықтайды, соның ішінде әртүрлі программалау тілдерінде жазылған бірнеше түрлі клиенттік қосымшаларға HTTP арқылы ресурстық жағдайларды қалай басқаруға және беруге болады.

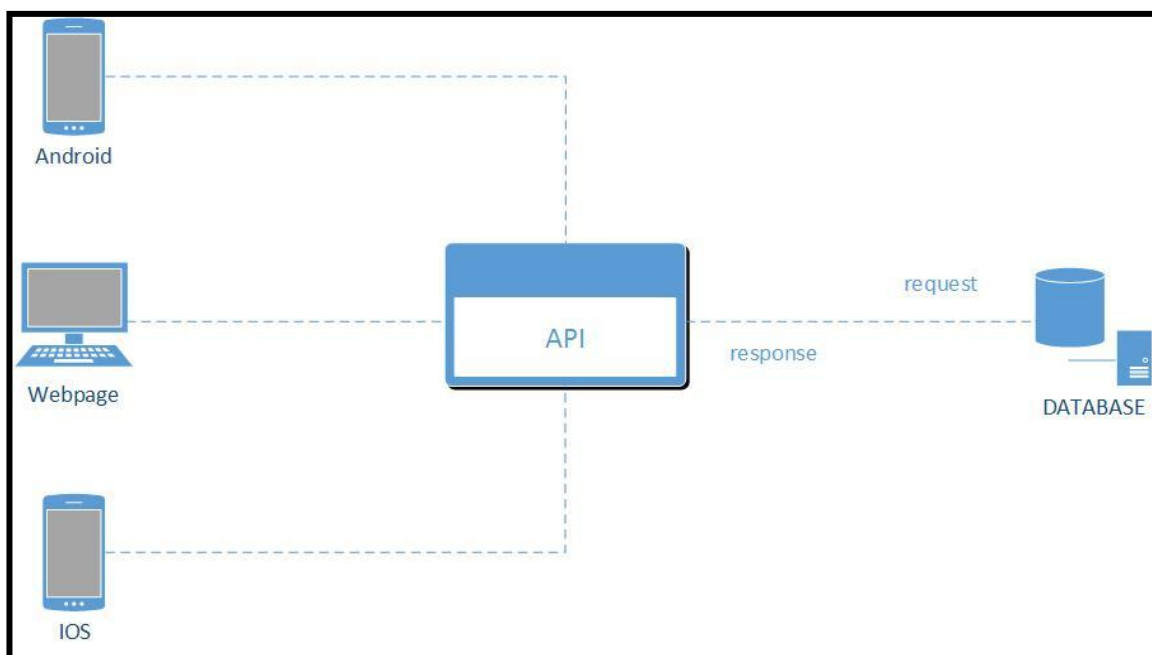
REST веб-қызметтерінің нақты іске асырылуы төрт негізгі принциптерге негізделеді:

- HTTP әдістерін анық пайдалану.
- сақталмаған жағдай.
- каталог құрылымына сәйкес URI кодын беру.
- деректерді XML немесе JSON
- екі пішімде тасымалдау.

RESTful веб-қызметінің басты ерекшеліктерінің бірі - RFC 2616-де анықталған хаттамаға сәйкес HTTP әдістерін нақты пайдалану болып табылады.

Интернетте қашықтағы сұраныстарды, әдеттегі HTTP-сұратуды (әдетте "Post" немесе "GET", мұндай сұрау "REST-сұрау" деп аталады) жіберуге болады және талап етілетін деректер сұрау параметрі ретінде беріледі.

Бұл технология барлық шектеулер мен шарттарды ескере отырып, "RESTful" деп аталады. REST технологиясы 1.1-суретте көрсетілген.



1.1-сурет – REST технологиясы

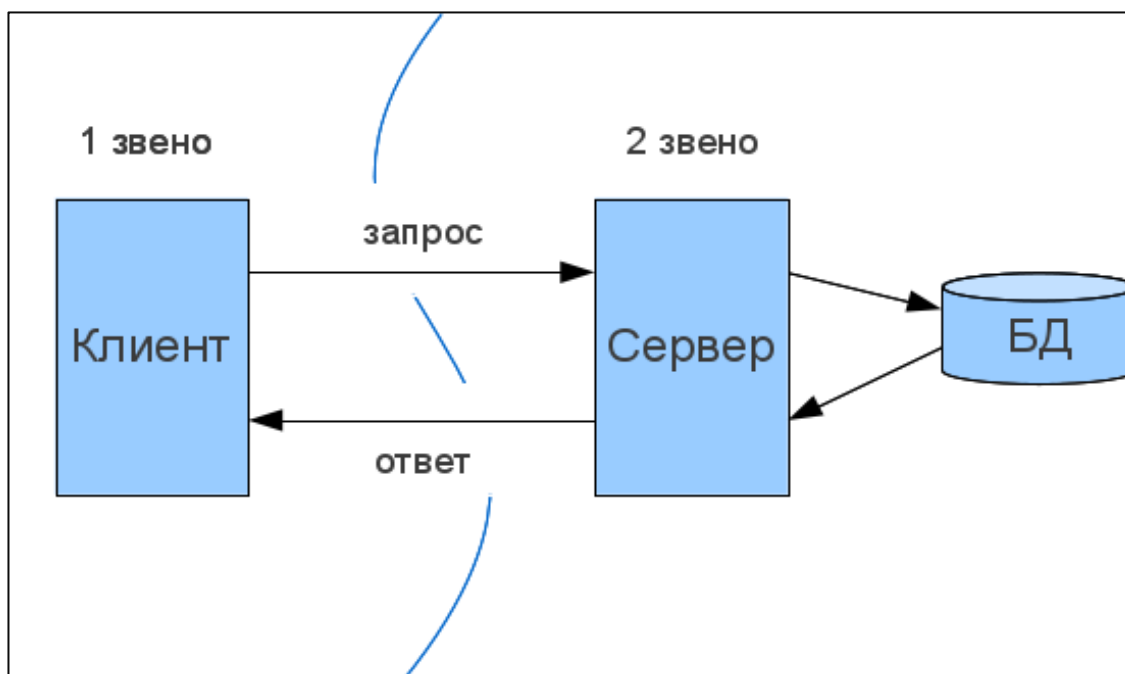
Rest негізгі мына мақсаттарды қамтиды

- компоненттердің өзара әрекеттесуінің масштабталуы
- интерфейстер ортақтығы
- компоненттерді тәуелсіз енгізу
- кідірісті төмендететін, қауіпсіздікті күшейтетін және ескірген жүйелерді

инкапсуляциялайтын аралық компоненттер

Rest архитектуралық стилі келесі алты шектеуді сипаттайды:

"Клиент-сервер" технологиясы – серверден клиенттер бірдей интерфейспен бөлінеді. Бұл жауапкершілікті бөлу, мысалы, клиенттер әрбір сервер үшін деректерді сақтау үшін жауап бермейді, сондықтан клиент кодының тасымалы жақсарады. Серверлер пайдаланушы интерфейсі немесе пайдаланушы күйі үшін жауап бермейді, сондықтан серверлер кеңірек және оңай болуы мүмкін. Серверлер мен клиенттер интерфейс өзгергенге дейін әзірленуі және ауыстырылуы мүмкін. 1.2-суретте клиент-сервер технологиясы көрсетілген.



**1.2-сурет – Клиент-сервер технологиясы**

Күйдің болмауы (Stateless) – кез келген клиенттен түскен әрбір сұрау оған қызмет көрсетуге қажетті барлық ақпаратты қамтиды, ал сессияның кез келген жағдайы клиентте сақталады. Сервер күйі болуы мүмкін; бұл шектеу сервер жағындағы күй ресурс ретінде URL арқылы жолдануын талап етеді. Бұл серверлерді мониторинг үшін анағұрлым көрінетін етіп қана қоймай, сонымен қатар желінің ішінара істен шығуы жағдайында оларды неғұрлым сенімді етеді, сондай-ақ олардың ауқымын қосымша жақсартады.

Кэштеу – клиенттер жауаптарды кэштей алады. Сондықтан жауаптар, анық немесе анық емес, клиенттің келесі сұрауларға жауап берген кезде ескі немесе жарамсыз деректерді пайдалануын болдырмау үшін өзін кэштирленетін немесе

болмауы тиіс. Жақсы басқарылатын кэштеу клиент пен сервердің кейбір қарым-қатынасын ішінара немесе толық жояды, әрі қарай масштабталу мен өнімділікті арттырады.

Көп деңгейлі жүйе – клиент қосылу жолы бойынша серверге немесе делдалға тікелей қосылатынын айқын анықтай алмайды. Сервердің делдалдары жүктемені теңгеруді қамтамасыз ете отырып және жалпы кэшти қамтамасыз ете отырып, жүйенің ауқымын жақсарта алады. Ол сондай-ақ қауіпсіздік саясатын сақтауды талап етуі мүмкін.

Талап бойынша Код (опциональды) – серверлер орындалатын логиканы бере отырып, клиенттің функционалдығын уақытша кеңейте немесе теңшей алады. Бұл мысал ретінде Java апплеттері және JavaScript сияқты клиенттік сценарийлер сияқты компиляцияланған компоненттер бола алады.

Бірыңғай интерфейс – төменде талқыланатын клиенттер мен серверлер арасындағы Бірыңғай интерфейс әрбір бөлікті өз бетінше дамытуға мүмкіндік бере отырып, архитектураны жеңілдетеді және бөледі. Мұндай интерфейснің төрт жетекші принципі төменде егжей-тегжейлі сипатталған.

### **1.3 Ұқсас жобаларға шолу**

Ақпараттық жүйелердің дамуы қазіргі уақытта университеттерде өзіндік сайттың немесе мобильдік қосымшаның болуын қажет етеді. Университет жайлы ақпараттарды, жаңалықтарды үйден шықпай ақ бақылап отыруға мүмкіншілігі бар. Қазіргі заманауи технологияларға сүйене отырып университет жайлы, ондағы аудиториялар жайлы ақпараттарды білу, сонғы жаңалықтарымен хабардар болу студенттер мен оқытушыларға және студент қонақтарына арналған бағдарламаның болуы барынша университет өмірін қолайластырады. Бұндай бағдарламалар санаулы. Соның бірі Пермск мемлекеттік университетінің «ПГНИУ карта» ақпараттық қосымшасы.

«ПГНИУ карта» қосымшасы бұл Пермск мемлекеттік университетінің студенттері мен қонақтарына университет аумағында бағдар жасап, компустар арасында қозғалуды ыңғайлы қылу үшін жасалған қосымша болып табылады. Бұл қосымшада қандай аудитория қай корпуста, сондай ақ университет бөлімдері яғни буфет, дәретхана қайда орналасқандығын білу қиындық тудырмайды. Бұл сервис студенттерге өзінің ұялы телефонының көмегімен ғылыми кітапхананың аудиториялары мен бөлімдерінің орналасуын білуге көмектеседі. «ПГНИУ карта» қосымшасының қарапайым және тартымды интерфейсін беттерді жүктеудің жылдам жылдамдығын қамтамасыз етеді, ал іздеу деген жолақта аудитория нөмерін дұрыс көрсетсе жеткілікті. Сұрау салғаннан кейін новигатор қандай аудитория қай корпуста орналасқандығын, ал картада шамамен орналасқан жерін көрсетеді.

«ПГНИУ карта» қосымшасының модульдік жүйе құрылымы:

– «аудитория табу» бұл бөлімде іздеу жолағына аудитория нөмерін жазып, сұрау салғаннан кейін сол аудиторияның қай корпусқа орналасқандығы туралы ақпаратты алуға болады. Аудитория орналасқан жерді картадан көрсетіп, сол аудитория жайлы ақпаратты көрсетеді. Картада аудитория орны визуалды түрде әдеміленіп көрсетілген.

– басқа бөлімдер. Оларға буфет, дәретхана және кітапхана жатады.

Бұл қосымшамен менің веб сайтымды салыстыратын болсақ. Қосымша Пермск мемлекеттік униерситетінің картасы болып табылады. Ал «Find System» қосымшасында университет туралы ақпаратты, соңғы жаңалықтарды және аудитория сабақ кестесін қарауға болады. Басты артықшылығы «Find System» жобасында аудиторияны брондау жүйесінің болуында. Брондау жүйесі оқытушыларға керекті аудиторияны керек сағатында брондауға мүмкіндік береді.



## **2 Жобалау бөлімі**

### **2.1 UML тілінде модельдеу**

Uniform Modeling Language (UML) – бағдарламалық қамтамасыз ету жүйесінің артефактілерін құжаттау және құрылымдау, баптау, визуализацияға арналған тіл.

UML тілі концептуалды моделін меңгеру маңызды, ол үш компонентті қамтиды: тілдің базалық құрылыс блоктарына арналған кейбір жалпы ережелер, олардың комбинациясы ережелері және жалпы тілдік механизм.

### **2.2 Rational rose бағдарламалық қамтамасы**

Rational Rose – Ақпараттық жүйелер және кәсіпорынды басқару үшін бағдарламалық қамтаманы жобалау және әзірлеуге арналған аспаптар. CASE басқа құралдары сияқты (ARIS, BPwin, ERwin), оны бизнес-процестерді талдау, модельдеу ге пайдалануға болады. Бұл өнімнің бірінші нұсқасы Rational Software шығарылды. Ары қарай IBM Rational Rose сатып алды.

Rational Rose басқа құралдардан принципті объектілі-бағытталған тәсілмен ерекшеленеді. Осы құралдың көмегімен жасалған графикалық модельдер объектілі-бағытталған принциптерге және UML ге негізделген. Rational Rose модельдеу құралдары әзірлеушілерге иерархияның әртүрлі деңгейлерінде бірге іс-қимыл мен бақылаудың барлық әсерлерін сақтай отырып, корпоративтік процестердің тұтас архитектурасын жасауға мүмкіндіктер береді.

Rational Rose артықшылықтары:

– бірлескен жұмысты қолдау. Бұл жағдай жобаның барлық қатысушылары үшін қарапайым қолдауды қамтамасыз етеді. Пайдаланушылар өз ортасын өзгертпей, өзінің бірегей үлгілерімен және жұмыс ортасымен жұмыс істей алады, сондай-ақ жалпы үлгілермен байланыс жасай алады.

– модельдерді басқару. Барлық модельдер оңай өзгеруі мүмкін. Бір модельдегі өзгерістер өзара байланысты үлгілерде автоматты түрде көрсетіледі. Нұсқалар және конфигурацияны басқару жүйесі модельдерді басқаруға қолданылады. Бұл кез келген бизнес-модельге оңай өзгерістер енгізуге мүмкіндіктер береді.

### **2.3 Прецеденттер диаграммасы**

UML диаграммаларының ішінде прецеденттер диаграммасы жүйе пайдаланушылары туралы мәліметтерді (актерлер ретінде белгілі) және олардың

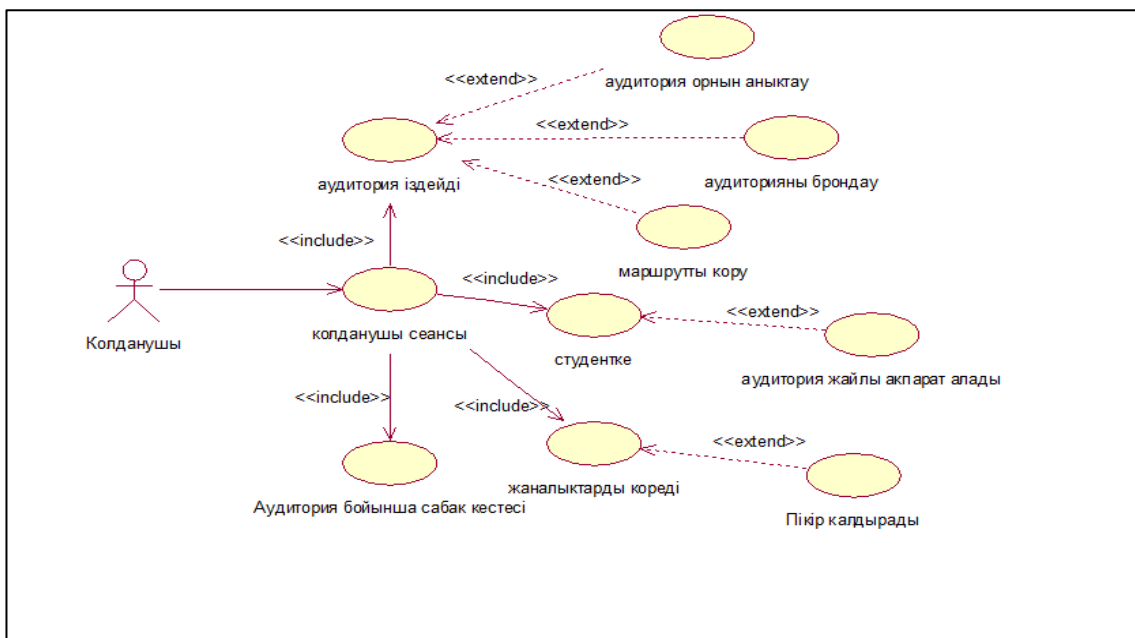
жүйемен әрекеті туралы ақпараттарды қосады. Оларды жасау үшін арнайы таңбалар қолданылады. Use Case диаграммасының эффективтілігі мыналарды ұсынады:

- жүйенің немесе қосымшаның адамдармен, ұйымдар мен сыртқы жүйелермен байланысу сценарийлері.
- жүйе белгілі бір мәнге жетуге көмектеседі.
- жүйе көлемі.

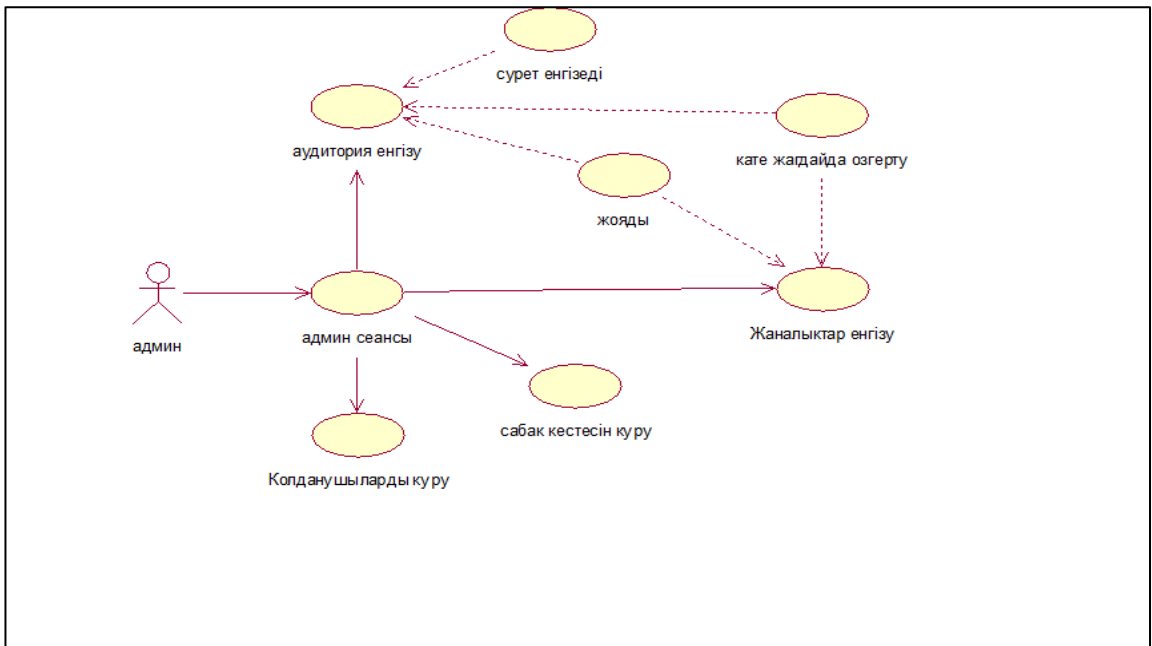
Прецеденттер диаграммасы ол қадамдар орындалатын тәртіпті көрсетпейді, сондықтан оны егжей тегжейлі диаграмма деп айта алмаймыз. Оның орнына бұл диаграмма қолдану сценарийлері, жүйе мен субъект арасындағы байланысты жақсы көрсетеді.

UML дегі прецеденттер диаграммасы мыналар үшін қолайлы:

- жүйенің пайдаланушымен өзара ара қатынасы туралы көріністі ұсыну
- жүйедегі функцияналдық талаптарды анықтап, оны ұйымдастыру
- жүйенің талаптары мен контексті көрсету



2.1-сурет – Қолданушының Use Case диаграммасы

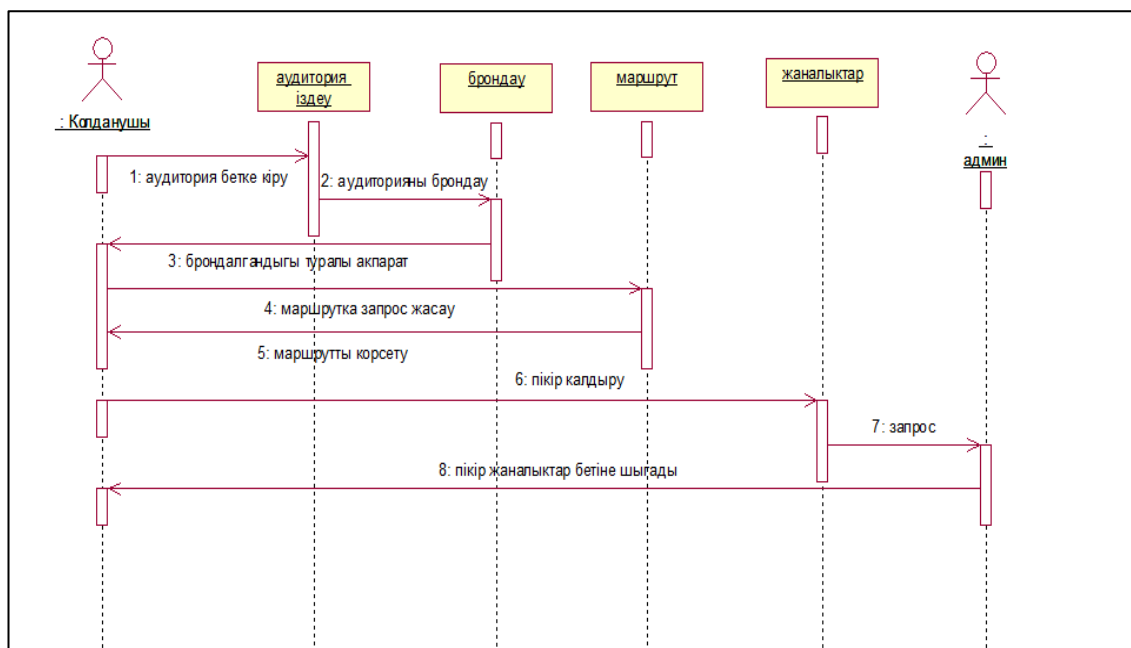


**2.2-сурет – Администратордың Use Case диаграммасы**

Прецеденттер диаграммасы – бұл белгілі әрекет етуші субъектіге (Actor) белгілі әсер бере алатын қолданбалы жүйе орындайтын іс-қимылдар тізбегінің сипаттамасы. Прецедент-бұл ең алдымен берілген құндылықтардың тәртіптік құрылымы, ал прецеденттер ынтымақтастық болған кезде бөлінеді.

## 2.4 Тізбек диаграммасы

Тізбек диаграммасы бұл ең көп қолданылатын өзара әрекеттесу диаграммасы болып табылады. Тізбек диаграммасы объектілер арасындағы байланысты белгілі бір ретпен, яғни өзара іс-қимыл орындалатын тәртіпте көрсетеді. Күй диаграммасы мен күй сценарий терминдерін тізбек диаграммасына сілтеме ретінде қолдануға болады. Тізбек диаграммалары жүйелік функциядағы объектілердің қандай тәртіпте орналасқандығын сипаттайды. Бұл диаграмманы қолданыстағы жүйелерге қойылатын талаптарды ұғыну үшін бизмендер мен бағдарламашылар арасында жиі қолданылады.



**2.3-сурет – Тізбек диаграммасы**

Тізбек диаграммасындағы белгілер:

– актерлер. UML диаграммаларындағы актер бұл жүйемен және олар арасындағы объектілер өзара әрекеттесетін рөл болып табылады. Біз пайдаланушылар мен басқа да субъектілерді белгілеу үшін актерды пайдаланамыз.

– өмір сызығы. Өмір сызығы бұл жеке пайдаланушыны бейнелейтін атаулы элемент. Тізбек диаграммасындағы әрбір дана өмір сызығымен көрсетілген. Өмір сызығының элементтері жоғарғы жақта тізбек диаграммасында көрсетілген. Өмір сызығының белгілену стандарты: Дана атауы: Класс атауы.

– хабарламалар. Объектілер арасындағы байланыс хабарламалар арқылы орындалады. Хабарламалар өмір сызығында ретті түрде пайда болады.

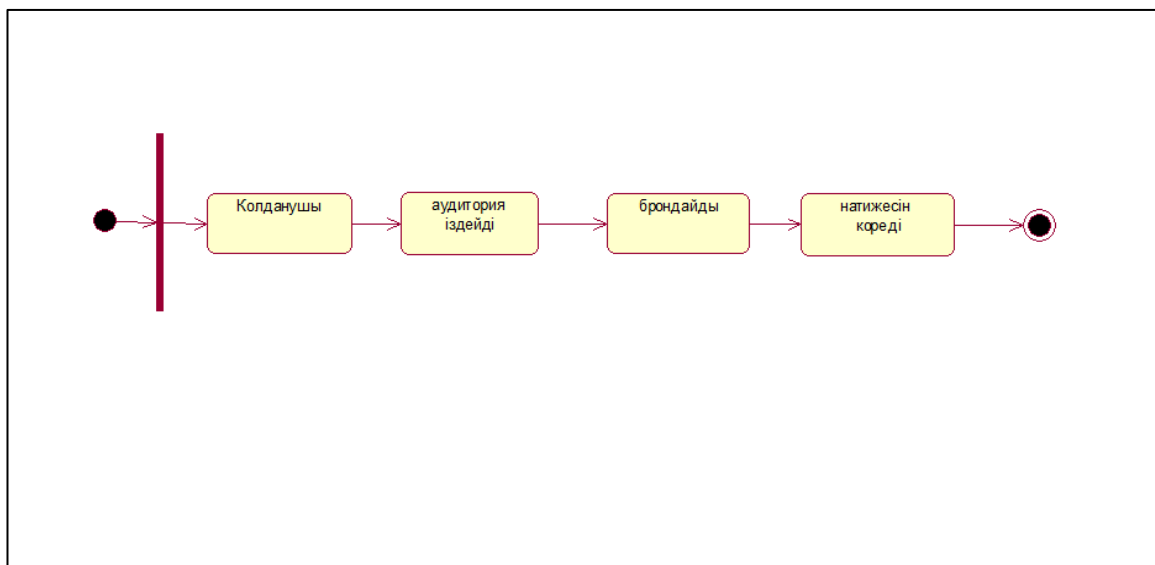
Өмір сызығы мен актердің айырмашылығы – Өмір сызығы жүйедегі ішкі объекті бейнелейді, ал актер жүйедегі сыртқы объекті бейнелеу үшін қолданылады.

## 2.5 Күй диаграммасы

Күй диаграммасы әр түрлі объектілер арасындағы ауысуларды көрсететін UML тіліндегі мінез құлық диаграммасы болып табылады.

Соңғы автомат – бұл белгілі бір уақыттағы объектінің күйін сақтайтын және өзгерте алатын кез келген құрылғы болып табылады. Әрбір күй диаграммасы бастапқы күйді көрсететін қара шеңберден басталады, қоршалған шеңбермен аяқталады, бұл аяқталған күйді білдіреді.

Күй диаграммасы көбінесе күй мен ауысуларды бейнелейді. Бұл диаграмма дөңгелектенген бұрыштары бар төртбұрыш түрінде көрсетілген.



**2.4-сурет – Колданушының күй диаграммасы**

Басқа UML диаграммалары сияқты күй диаграммасы да бірнеше қолданыстарға ие:

- реактивті жүйеді оқиғаларды басқаратын нысандарды бейнелеу.
- бизнес контексте қолданылған сценарийлерді иллюстрациялау.
- объект өзінің өмірлік жолында қандай күй кешеді соны сипаттау.
- соңғы автоматтың жалпы мінез құлқын көрсету.

Объектілердің жай – күйін модельдеу күй диаграммасы арқылы жүзеге асырылады.

## 2.6 Класстар диаграммасы

Класстар диаграммасының мақсаты:

- жүйедігі классификатордың статикалық құрылымын көрсетеді.
- UML да жазылған басқа да құрылымдық диаграммалар үшін басты белгілермен қамтамасыз етеді.
- бағдарламалаушылар үшін пайдалы.
- бизнес талдаушылар бизнес тұрғысынан модельдеу үшін класстар диаграммасын қолданады.

Класс дегеніміз жүйедегі ұқсас рөлдері бар объектілердің сипаттамасы.

- структуралық ерекшеліктер (атрибууттар) класс объектісінің нене білетіндігін анықтайды.

Класс объектісінің күйін көрсету.

– кластың құрылымдық немесе статикалық қасиеттерін сипаттау.

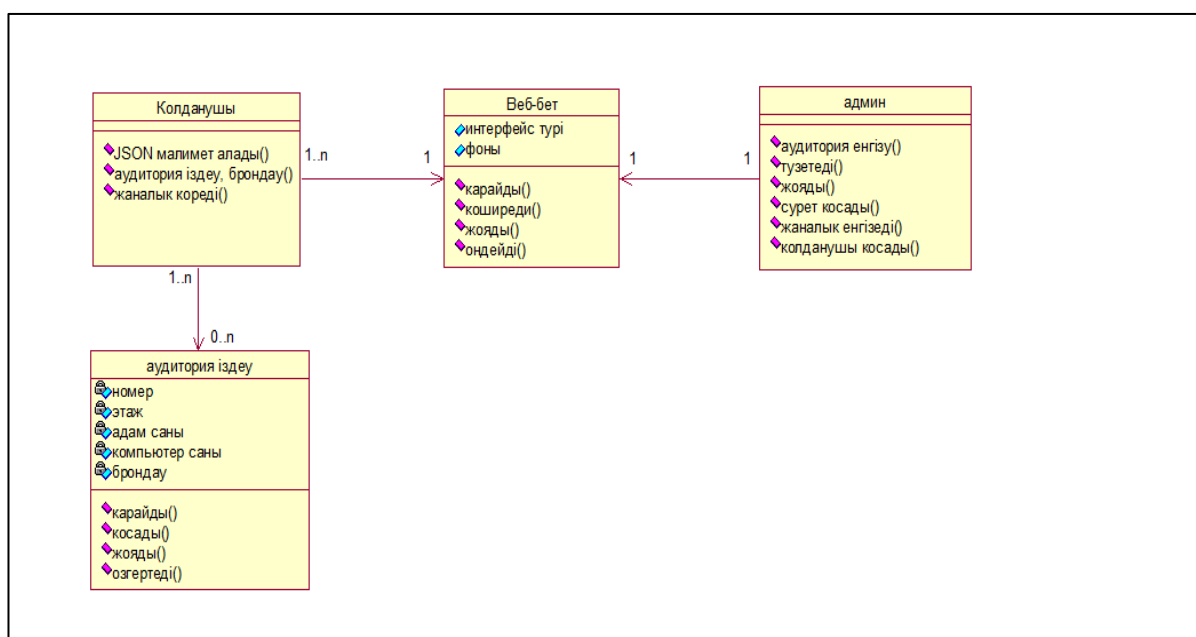
– мінез құлық ерекшеліктері класс объектісінің не істей алатындығын анықтайды.

Классты белгілеу үш бөліктен тұрады. Олар:

1. класс аты. Класс аты бірінші бөлікте жазылады.

2. класс атрибуты. Класс атрибуты екінші бөлікте жазылады. Атрибуттың типі қос нүктеден кейін белгіленеді. Атрибуттар кодта айнымалы мүшелерді бейнелейді.

3. класстармен операциялар. Яғни методдар. Операциялар үшінші бөлікте көрсетіледі. Метод қайтару сигнатурасы қос нүктеден кейін орындалады. Операциялар кодта класс методтарын көрсетеді.



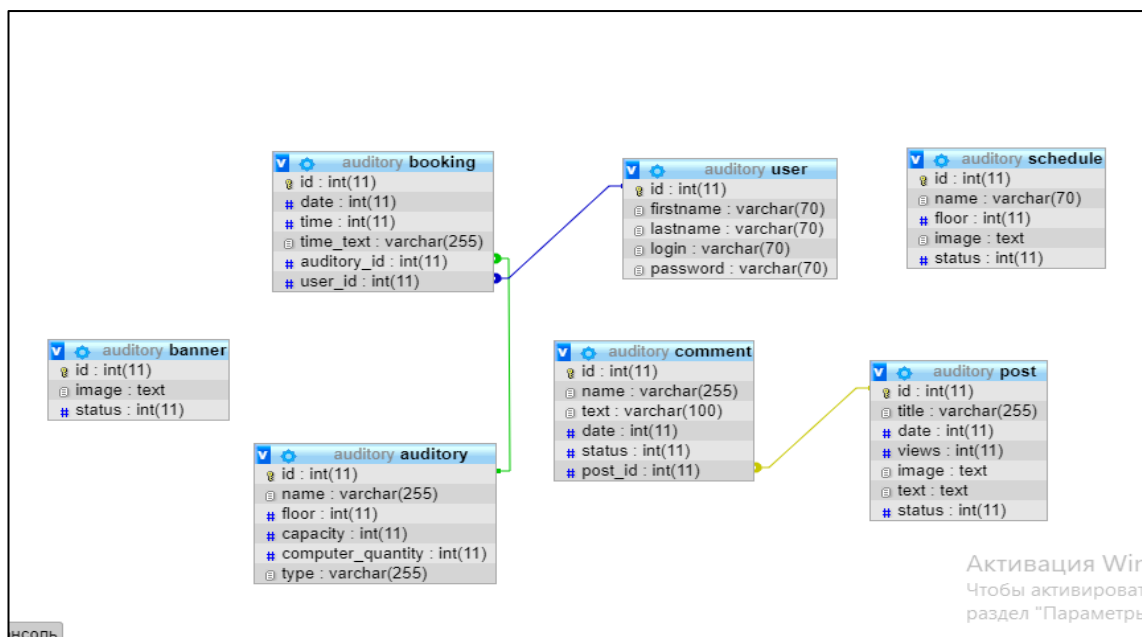
2.5-сурет – Қолданушының класстар диаграммасы

Диаграмманың бұл түрі сипатталған кластардың базалық құрылымы негізінде логикалық көріністі құруға мүмкіндік береді. Диаграммалар класстар арасындағы өзара байланысты көрсетуге мүмкіндік береді. Rational Rose диаграмманың осы түрін пайдалана отырып түрлі нотациялар құруға мүмкіндік береді.

## 2.9 ER диаграмма құру

ER диаграмма бұл деректер моделі. Кестелер арасындағы байланысты көрсетеді. Деректер қорын жобалағанда ER модельін қолданады. Бұл негізгі байланыстарды көрсетуді визуальді түрде көрсетеді. Модельді алғаш ұсынған

Питер Чен. Алғаш рет 1976 жылы жарыққа шықты. Проектінің ER диаграммасы 3.8-суретте көрсетілген.



2.9-сурет – Проектінің ER диаграммасы



## 3 Қолданылған бағдарламалық қамтамалар

### 3.1 MySQL технологиясы

MySQL – әлемдегі ең кең қолданылатын деректер қорының бірі. MySQL қолданысқа өте ыңғайлы деректер қоры болып табылады және жоғары жылдамдықта жұмыс жасайды. MySQL Linux бағдарламасы шыққаннан кейінгі ең мықтысы болып табылады. MySQL жобалаудың ең негізгі артықшылығы жылдамдығы, сенімділігі және пайдалану қарапайымдылығы.

MySQL – бұл құрылымдалған сұраныстар тіліне (SQL) негізделген Oracle қолдайтын ашық кодты реляциялық басқару жүйесі деп сипаттайды. MySQL деректер қоры SQL (Structured Query Language) ANSI 92 стандарты тілінде кез келген сұрау жасай алады. SQL MySQL сұрау тілінің бір бөлігі ретінде, сондай-ақ Деректер қорына кіру үшін қолданылатын және ең жиі қолданылатын стандартты тіл ретінде сипаттауға болады. MySQL-ашық бастапқы коды бар бағдарламалық қамтама болып табылады.

MySQL-тың қысқаша ерекшеліктері:

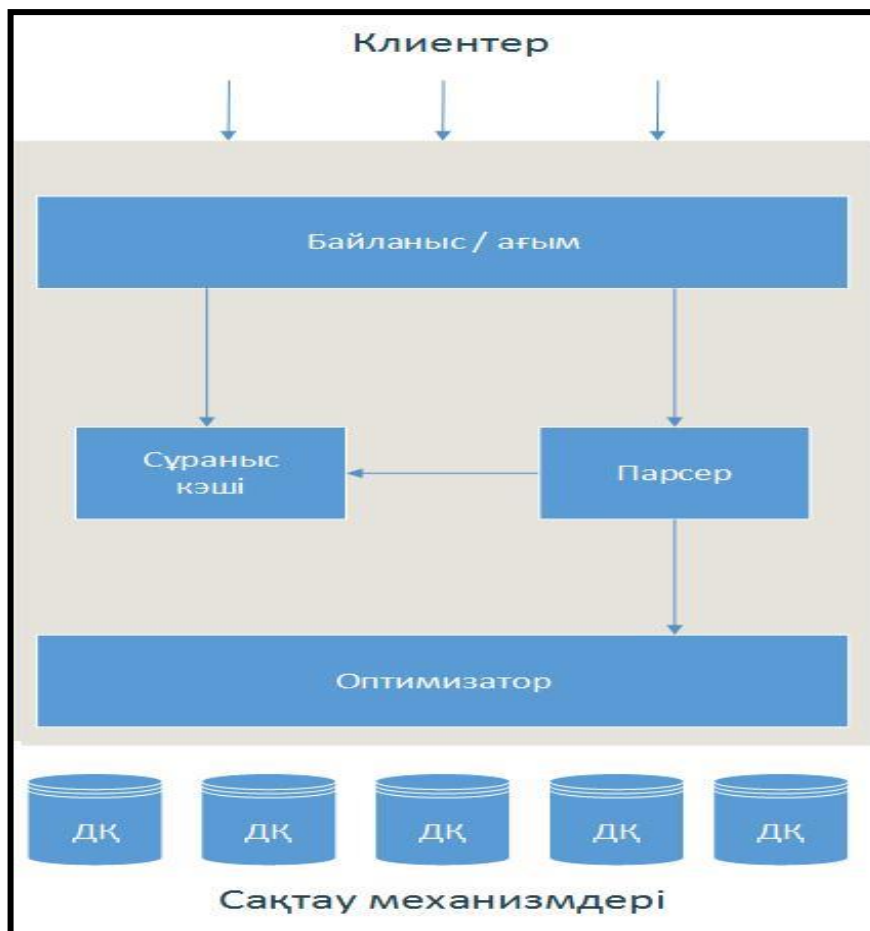
- бір уақытта жұмыс істейтін жұмысшыларға дерекқорға қосылуға мүмкіндік береді;
- 50 миллион санақ кестесіне жетуі мүмкін;
- бұл MySQL командаларын орындау кезінде Servers-дағы ең жылдам дерекқорлардың бірі;
- қарапайым және жақсы қалыптасқан қауіпсіздік жүйесі;
- пайдаланушы кез-келген күрделілік сұрауларын жасай алады. Мысалы, пайдаланушылардан `SELECT * FROM WHERE` пайдаланушы аты `IN` (хабарлардан `SELECT id`).

Деректер қорын басқару жүйесі PhpMyAdmin қосымшасын орындалады. Орташа баға бойынша MySQL деректер қоры бірнеше гигабайт жады мен бірнеше процессор қажет қуатты аппараттық серверлерде үлкен масштабты болуы мүмкін.

MySQL-lamp деп аталатын ашық бастапқы коды бар корпоративтік стектің маңызды бөлігі. LAMP-бұл Linux операциялық жүйе, Apache веб-сервері, MySQL реляциялық деректер қорын басқару жүйесі және PHP-скрипттердің объектілі-бағытталған тілі ретінде пайдаланылатын веб-әзірлеу платформасы (Кейде PHP орнына Perl немесе Python қолданылады).

MySQL клиент-сервер моделіне негізделген. MySQL ядросы MySQL сервері болып табылады, ол деректер қорының барлық нұсқауларын (немесе командаларын) өңдейді. MySQL сервері желілік ортада клиент-сервер қолдану үшін жеке бағдарлама ретінде, сондай-ақ жеке бағдарламаларға кіріктірілуі (немесе байланысты) мүмкін кітапхана ретінде қолжетімді. MySQL басқару логикасын сақтау процедуралары түрінде сақтай алады, сондай-ақ деректердің бүтіндігін бақылай алады, олардың дұрыстығын және т.б. тексереді. Сонымен қатар, MySQL, PostgreSQL, Oracle және т.б. сияқты көптеген PCУБД

қолданылады. Алайда тәжірибе көрсеткендей, MySQL көбінесе Php-мен бірге қолданылады.



3.1-сурет – MySQL архитектурасы

MySQL деректерді сақтауға және оларға InnoDB, CSV және NDB қоса, бірнеше сақтау механизмдері арқылы кіруге мүмкіндік береді. MySQL сондай-ақ ұзақ мерзім мен өнімділікті арттыру үшін деректер кестесін көрсетеді.

Ақпаратты сақтау-маңызды компьютерлік функциялардың бірі. Ең жиі қолданылатын құрал-деректер қоры болып есептеледі. Деректер қоры дегеніміз құрылымдағы ақпаратты қамтитын арнайы форматтағы файл болып табылады. ДҚ-де сақталатын деректер кесте түрінде дайындалады.

Иерархиялық ДБ жазу элементтерін екіншісінен кейін жазады және олардың біреуі негізгі, ал екіншісі - бағынышты элементтер болып табылады. Мұнда элементтер белгілі бір тізбектегі қадаммен түзетіледі. Бұл хабар-ошарсыз кеткендерді іздеу бағытында жүзеге асырылады. Мысалы: Windows файлдық жүйелері, интернет-ресурста каталогтар жиынтығы.

## 3.2 JavaScript бағдарламалау тілі

JavaScript HTML құжатында қолданылатын және вебсайттарда динамикалық интерактивтілікті қамтамасыз ететін толық функционалды бағдарламалау тілі. Ол Mozilla, Mozilla корпорациясының негізін қалаушы Брендан Айч пен құрылды.

JavaScript керемет әмбебап. Шағыннан бастау алады, мұндай карусельдер сияқты қарапайым функциядан, суреттер галереясы, өзгертілетін макеттер мен түймелерді басуға жауапты болады. Тәжірибеге ие бола отырып, 2D анимацияланған және 3D графикаларын, толық масштабты деректер қоры бар қосымшаларды және т. б. жасай алады. JavaScript өзі өте ықшам, бірақөте икемді. Әзірлеушілер JavaScript негізгі тілінің үстінен көптеген құралдар жазды, олар өте аз күшпен қосымша функциялардың үлкен санын бұғаттайды.

Оларға жатады:

- html динамикалық жасау және CSS стильдері, бейне ағындарын басып алу және басқару, пайдаланушының веб-камерасымен жұмыс істеу немесе 3D-графика мен аудиосемплдар жасау сияқты түрлі функционалдық мүмкіндіктерді ұсынатын браузерлерге арналған API-қолданбалар интерфейстері (API).

- үшінші тарап API әзірлеушілерге Twitter немесе Facebook сияқты басқа әзірлеушілерден өз сайттарына функционалдылықты біріктіруге мүмкіндік береді.

- html-ге үшінші тарап фреймворктерін және кітапханаларын қолдана аласыз, бұл сайттар мен қосымшаларды жасауды жеделдетуге мүмкіндік береді.

Алғашында JavaScript құрылған кезде оның бастапқы аты «LiveScript» болды. Сол кезде Java тілі танымал болғандықтан, маркетингтік бұл тілге жаңа атау беруді ұсынды. Осылайша «LiveScript» тіркесіне Java сөзі қосылып «JavaScript» деп аталды.

JavaScript тек қана браузерде ғана жұмыс жасау емес, сонымен қатар барлық жерде жұмыс жасау үшін арнайы интерпретатор программасы болса жеткілікті. Скриптті орындау процесі "интерпретация" деп аталады.

JavaScript бет манипуляциясына, келушімен және сервермен өзара әрекеттесуге қатысты барлық нәрсені жасай алады:

- жаңа HTML тегтерін жасау, бар элементтерді жою, стиль элементтерін өзгерту, элементтерді жасыру және т. б.

- келушінің іс-әрекетіне, тінтуірге, жүгіргіні, пернетақтаны және т. б. жылжытуға әрекет ету. нұқыңыз.

- серверге сұраныстарды жіберу және бетті қайта жүктеусіз деректерді жүктеу (бұл технология "AJAX" деп аталады).

- cookie таңдау және алу, деректерді сұрау, хабарларды басып шығару.

- JavaScript-бұл клиент жағында да, сервер жағында да орындалатын сценарийлерді жазу үшін объектілі-бағытталған бағдарламалау тілі.

```
1 <!DOCTYPE HTML>
2 <html>
3
4 ▼ <head>
5     <!-- Кодировканы анықтайтын мета тэг -->
6     <meta charset="utf-8">
7 </head>
8
9 ▼ <body>
10
11     <p>Документтің басы...</p>
12     <script>
13         alert( 'Сәлем, Әлем!' );
14     </script>
15
16     <p>...Документтің соңы</p>
17
18 </body>
19
20 </html>
```

### 3.2-сурет – JavaScript бағдарламалау тілінде «Hello world» коды

Осы тілді пайдаланып, көрінетін элементтермен сонымен HTML құжаттарын да динамикалық басқаруға болады.

JavaScript сізге құжаттамамен әрекеттесуге мүмкіндік береді. Document нысаны және оның ішіндегі Нысандар HTML-құжаттамасының бөлімдерімен оқу және кейде өзара әрекеттесуге мүмкіндік береді. Мәтінді өздігінен оқи алмайды, бірақ, мысалы, осы құжаттардағы гипермәтін сілтемелерінің тізімін алуға болады.

JavaScript сізге пайдаланушы әрекеттестігіне мүмкіндік береді. Тілдің маңызды ерекшелігі-нақты жағдайлар (әдетте пайдаланушының мінез-құлқында) туралы сөз болған кезде кодтың алдын ала бөліктерін анықтау мүмкін болған жағдайларда әзірлеушілерді сәйкестендіру мүмкіндігі. JavaScript кез келген алдын-ала анықталған функцияларды шарт жасаушылары ретінде пайдалана алады. JavaScript жақын математикалық задачаларды орындауға мүмкіндік береді. Сонымен қатар, бұл тіл күн мен уақыт мәндеріне арналған құралдарды әзірлейді. JavaScript шын мәнінде CGI бағдарламасының баламасы және Perl сценарийлерінің тіліне, сондай-ақ Java қосымша толықтыруларына балама болып табылады.

### 3.3 HTML тілі туралы түсінік

HTML (ағылш. "hyper text markup language" – гипермәтіндік белгілеу тілі) бұл Интернетте сайттарды жасау кезінде қолданылатын арнайы белгілеу тілі болып есептелінеді. HTML құжаттары бетбелгі дескрипторы немесе тег деп аталатын арнайы кодтық сөздерді, мәтінді пішімдеу, құжат құрылымын құру,

сілтемелер мен мультимедиалық нысандарды енгізу үшін қолданады. Браузерлер html-ді жақсы түсінеді және оны түсінікті түрде түсіндіре алады. Html белгілеу тілі ретінде жақсы танымал болды. Қазіргі уақытта бұл сайтта белгіленетін жалғыз тіл. Ол интерактивті беттерді жасауға мүмкіндік беретін "қозғалғыш" деп аталатын басқа бағдарламалау тілінде (негізінен, php) жасалған (html бұл мүмкіндік бермейді).

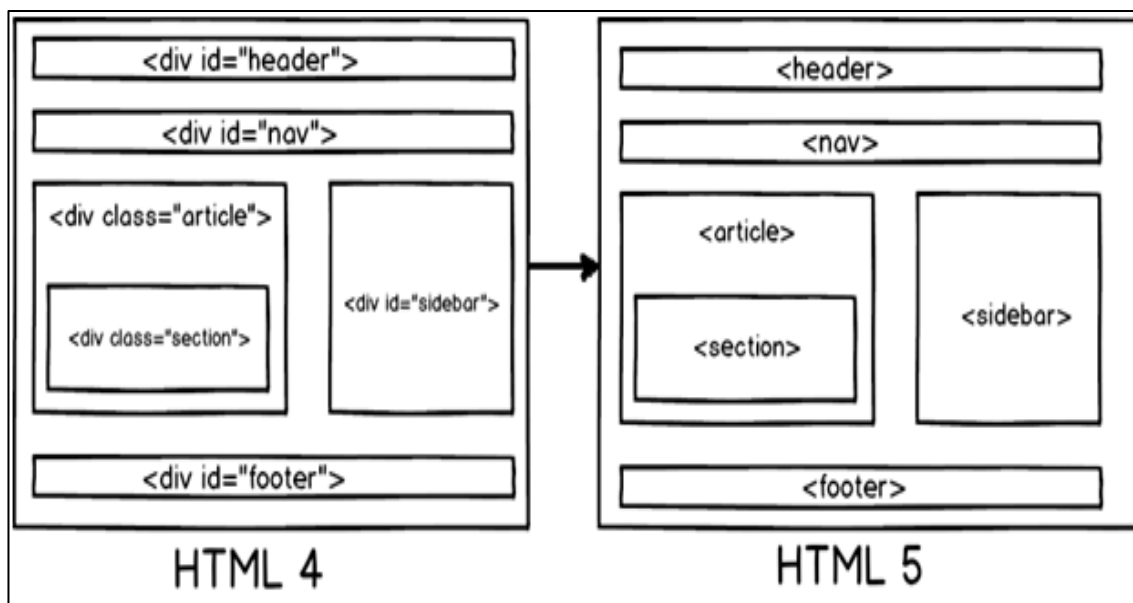
Html-да жасалған файлдар.html немесе .htm кеңейтілімдерімен көрсетіледі.

Бұл веб-беттегі қандай элементтердің бар екендігі туралы ақпаратты браузерге хабарлай алатын тіл. Кез келген тіл сияқты, HTML-да сөздер (командалар) және осы сөздерді жазу ережелері (тіл синтаксисі) бар.

Төмендегі HTML коды мысалында көрсетілгендей, ол әрқашан `<!DOCTYPE html>` тан басталады. Браузер үшін бұл элементтің пайда болуы HTML құрылымын көрсетеді:

```
<!DOCTYPE html>
<html>
<body>
<h1>мұнда тақырып орналасады</h1>
<p>ал бұл жерде бірінші абзац болады</p>
</ body>
</ html>
```

Негізінен, HTML тілі веб-беттерді жасау үшін қолданылады. HTML5 және HTML 4 арасындағы айырмашылықтар 2.3-суретте көрсетілген.



**3.3-сурет – HTML 5 пен HTML 4 тің айырмашылықтары**

HTML тілі "гипермәтін белгілеу тілі" деп аударылады. Бұл тіл веб-беттерді жасау кезінде негізгі бағдарламалау тілі болып табылады және белгілі бір командалардың көмегімен беттің әр түрлі (мәтін, графика, кестелер және т.б.) элементтерін орналастыру үшін пайдаланылады.

### 3.4 Apache HTTP сервері

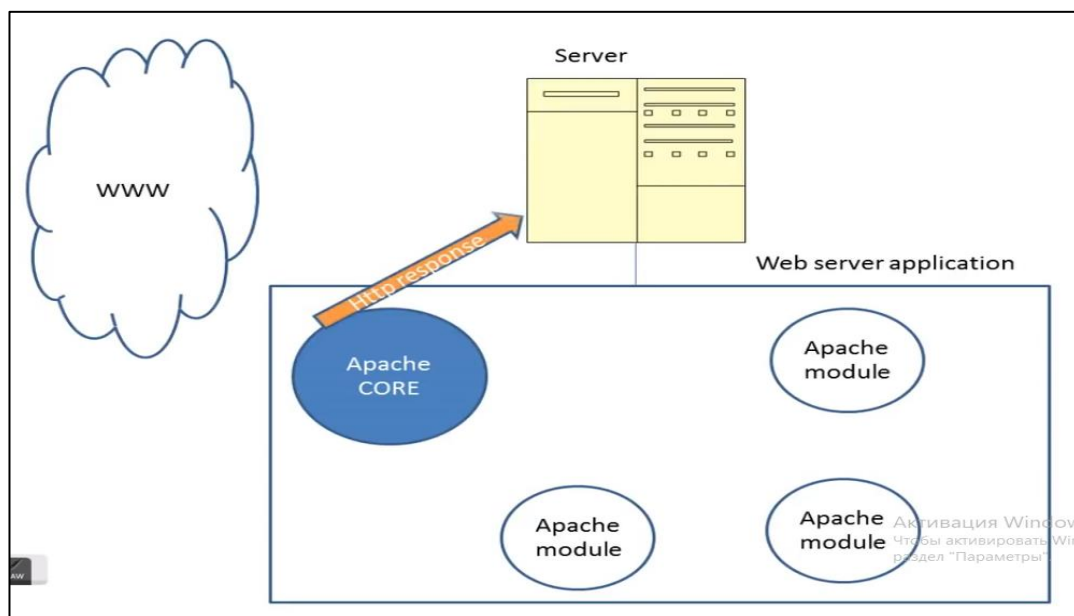
Apache – бұл өте жақсы қауіпсіздікке ие HTTP сервер. HTTP сервер деп біз HTTP сұрауларды өңдеуге арналған бағдарламалық қамтаманы айтамыз. Apache веб – серверінің басты қызметі HTTP сұрауларға жауап қайтару және сол сұрауларды өңдеу және беттің динамикалық мазмұнын генерациялау. Икемділік .htaccess файлын қолдану арқылы қол жеткізіледі, Apache серверінің жаһандық баптауларын қолдануға болады. Apache өзінің жаһандық баптауларына ие, бірақта ол сайтқа сұрау салғанда ол .htaccess файлын іздеуді, директиваларды оқиды, олар жаһандыққа жатпайды.

Apache виртуалды хостарды анықтайтын механизмге ие, сондықтан бір IP адрессте шексіз сайттар болуы мүмкін. Сонымен қоса, көптеген бағдарламалау тілдерімен жұмыс жасауға мүмкіндік беретін көптеген модульдер бар.

Бүгінгі күнде Apache ең танымал веб сервер ге арналған бағдарламалық қамтама. Apache веб – сервері кроссплатформалық болып табылады. Яғни Windows, Linux, IOS сынды операциялық жүйелерде орындала береді.

Apache мына қызметтерді атқарады:

1. клиенттен сұрау алу
2. клиенттен келген сұрауды өңдеуді
3. клиент сұрауында көрсетілген ресурсты іздейді
4. сервер клиентті қалай түсінгендігі туралы және клиент көрсеткен әрекетті орындай алдыма сол туралы күй кодымен жабдықталған HTTP жауап қайтарады.



3.4-сурет – Apache веб-сервері

Apache сервері өте ыңғайлы, өйткені модульдік құрылымы бар. Модульдер сервер әкімшілеріне қосымша мүмкіндіктерді қосуға немесе өшіруге мүмкіндік береді. Apache серверінің қауіпсіздігі, кэштеу, URL өңдеу, пароль арқылы аутентификация және т.б. Модульдер бар. Файл арқылы өз конфигурациясын теңшеуге болады. Сонымен қатар, әдепкі бойынша htaccess және Hostinger параметрлері қолданылады.

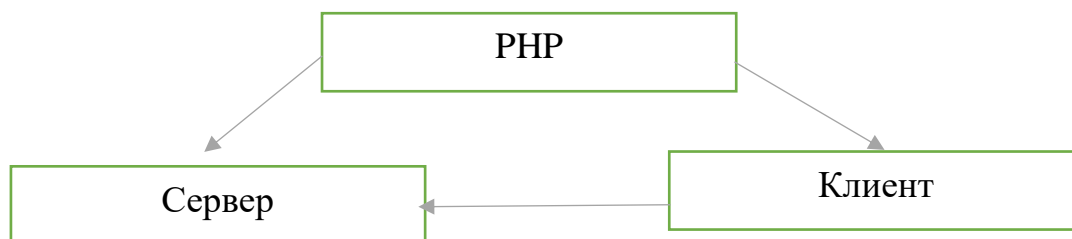
### 3.5 PHP программалау тілі

PHP тілінің тарихы 1995 жылдан бастау алады. Дәл осы жылы бағдарламашы Расмус Лерфорд Perl/CGI сайтқа кіруші қолданушылардың санын есептейтін сценарий жазып шықты. PHP тілінің алғашқы аббревиатурасы Personal Home Page деп оқылған, ал қазір ол гипертексттік препроцессор деп оқылады. PHP бұл серверлік платформааралық HTML-ге енгізілген сценарий жазу тілі болып есептелінеді.

Мұнда серверлік дегенде, PHP-де орындалатын операциялар сервер тарабында орындалады. Сервер – арнайы компьютер. Серверде веб беттер болады.

Платформааралық дегені PHP тілі Linux, Windows, Macintosh, OS/2 және т.б серверлік операциялық жүйелерде қолданыла береді. PHP сценарийлерін операциялық жүйеде құруға болмайды. Бұл HTML кодының ішіне ендірілуі мүмкін. PHP ді басқа тілдермен салыстырғанда бұл тіл өте ыңғайлы және қолдануға жақсы болып келеді. Php тілінің баламаларына мыналар жатады: HTML, CGI, ASP, JSP. JavaScript PHP тілінің баламасына жатпайды, себебі ол клиенттік технологияға жатады. Қазіргі кезде неліктен барлық бағдарламашылар PHP тілін пайдалануда? Өйткені бұл тіл CGI сценарийлерге қарағанда оңай және жылдам болып табылады, программалар жылдам орындалады.

PHP-ді серверлік тіл. Код серверлік тарапта орындалып, браузерге веб беттерді жібереді.



PHP тілінің негізгі атқаратын қызметі:

- бұл тілдің мақсаты веб бағдарламашыға веб беттерді әзірлеуді жеңілдету
- жүйелік файлдарды орындау, файлдарды құру, ашу, оқу және жабу, сонымен бірге каталогтарды құру.



– деректерді файлда сақтау, деректерді электронды почтадан жіберу, өңделінген мәліметті қолданушыға қайтару болып табылады.

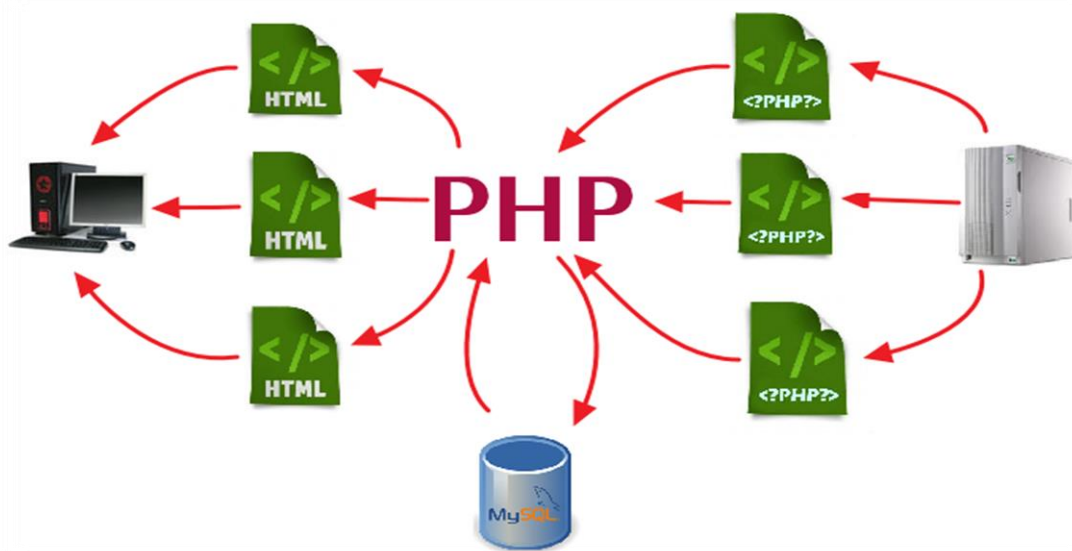
– деректер қорына элементтерді қосуға, жоюға және өзгертуге рұқсат алу үшін.

- қолданушының тіркелген аты мен құпия сөзін тексеру үшін.
- бейне суреттер құру үшін.
- деректерді шифрлеуге.

PHP бағдарламасымен жұмыс жасау үшін Апачи сервері мен MySQL деректер қоры бар программа пакеттерін қондыру қажет. PHP тілінің базалық құрылымына (Perl ұқсас) қосымша кластар кітапханасы стандартты функциялардың кең спектрін және өндіруші үшін үлкен артықшылықтарды ұсынатын C-бағдарламалау тіліне ұқсас. Біздің хостинг тарифтеріміз PHP4 немесе PHP5 нұсқаларына қолдау көрсетеді. Жылдамдығын салыстыру PHP 7.0 PHP 5.6 суретте көрсетілген 3.5.

PHP – бұл бағдарламалау тілінде қолданылатын сервердің бөлігі болып табылады. PHP дің ұзақ тарихы бар – оның алғашқы нұсқасы 1995 жылы, 20 жыл бұрын пайда болған.

Бірінші негізгі PHP 7 осы өнімділік туралы айтады: PHP 7 HHVM өнімділігі өнімділікке жақын (PHP кодын Facebook-ке бірнеше рет түрлендіруді жылдамдатуға мүмкіндік беретін жүйе). кем дегенде, осы себепті Zend технологиясының негізін қалаушы (PHP негізгі әзірлеуші компаниясы). Php тілі тек HTML құжаттарын ғана емес, JPG, PNG, GIF файлдарын да жасай алады. PHP тілі MySQL деректер қорымен қарым қатынас жасайды. MySQL бұл интернет сайттарды жасаудың оңтайлы тәсілі болып табылады.



3.5-сурет – PHP тілінің жұмыс істеу сұлбасы

PHP негізгі артықшылықтарының бірі—деректер қорының кең спектрін қолдау. Мәліметтер қорын қолданатын скрипт жасау өте оңай. Сонымен қатар, бұл мәліметтер қорымен бөлек мәліметтер қорына (мысалы, mysql) немесе PDO сияқты мәліметтер қорынан абстрагирлеу, немесе ODBC кеңейтуі арқылы Open Source Database (ODBC) мәліметтер қорынан қолдау тапқан кез келген мәліметтер қорына тән кеңейтулерді пайдалануға болады. URL немесе сокетті CouchDB сияқты басқа дерекқор үшін пайдалануға болады.

### 3.6 YII2.0 тірегі

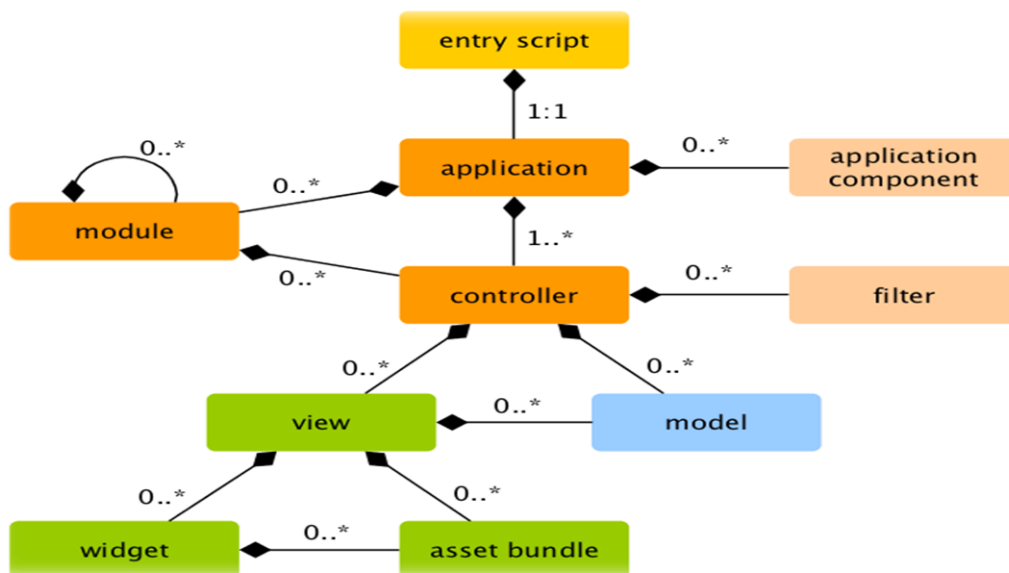
#### Артықшылығы

Фреймворк пайдаланудың кейбір айқын артықшылықтары:

- проблемаларды шешудің стандартты әдістерін пайдалана отырып, код дисбалансын азайтады немесе жояды.
- Формалар мен қауіпсіздікті тексеру сияқты ретті міндеттерге кететін уақытты қысқарту.
- бұл стандарттарға сай командалық жұмысты жеңілдетеді.
- бұл жалпы архитектурасы мен әдістерін пайдалана отырып, кодты қолдауды жеңілдетеді.
- фреймворкаларды, жалпыланған міндеттер мен жаңа мүмкіндіктерді қолдайтын белсенді әзірлеушілер қауымдастығы бар.

#### Кемшілігі

Фреймворк қолданудағы кемшіліктер бар. Біріншіден, құрылымы бар бүкіл код серверге жүктемені жасайды және бұл нақты проблема болуы мүмкін. Бақытымызға орай, жүктеу әсерін азайтуға мүмкіндік беретін кэштеу опциялары бар, ал корпоративтік бағдарламалар үшін Сіз сұрау уақытын азайту үшін өз SQL пайдалана аласыз. Сондықтан серверді қайта жүктеу мәселесі бар фреймворк пайдалану сізді тоқтатпауы керек.



3.6-сурет – Yii 2.0 тірегінің архитектурасы

## **4 Веб бетті құру және іске асыру**

### **4.1 «Find System» проектісі туралы**

«Find System» проектісі Restful Api технологиясын қолдана отырып жасалды. «Find System» проектісі көпфункциональды веб сайттан және мобильді қосымшадан тұрады.

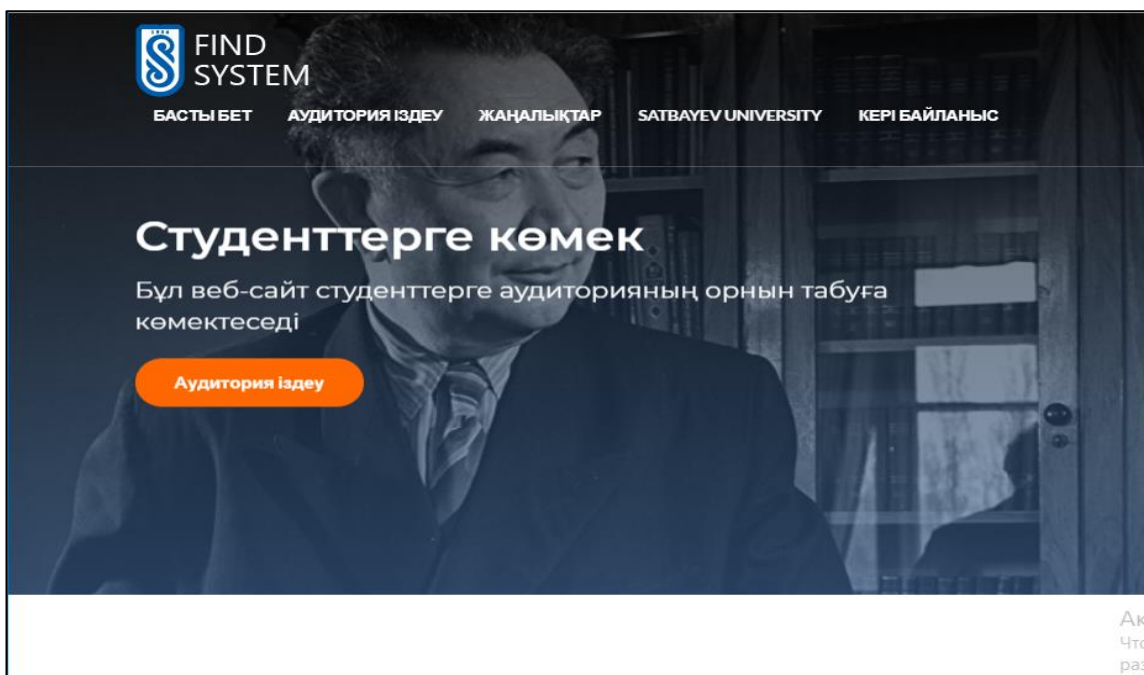
Restful API бұл веб сайтпен мобильді қосымша арасындағы байланысты үздіксіз қамтамасыз ете отырып, ақпарат алуды ұйымдастырады. Бұл веб беттің негізгі қызметі университетіміз оқытушылары мен студенттерге аудитория орнын табуды жеңілдету болып табылады. Ал мобильді қосымшаны барлық қолданушылар қолдана алады.

Веб сайт бұл админ беттен және басты беттен тұрады. Бұл веб сайт арқылы студенттер университетіміздің тарихымен танысып соңғы жаңалықтардан хабардар бола алады. Сонымен бірге аудитория жайлы ақпаратты көре алады. Аудитория бөлімінде мұғалімдерге және студенттерге деген екі бөлім бар. Мұғалімдер аудиторияның адам санына қарай кабинет таңдай алады және қай аудиторияда қанша компьютер бар соған қарай кабинет таңдай алады. Және сол аудиторияны өзі қалаған уақытқа брондау жүйесі қарастырылған. Студенттер бөлімінде студент аудитория жайлы ақпаратты біле алады.

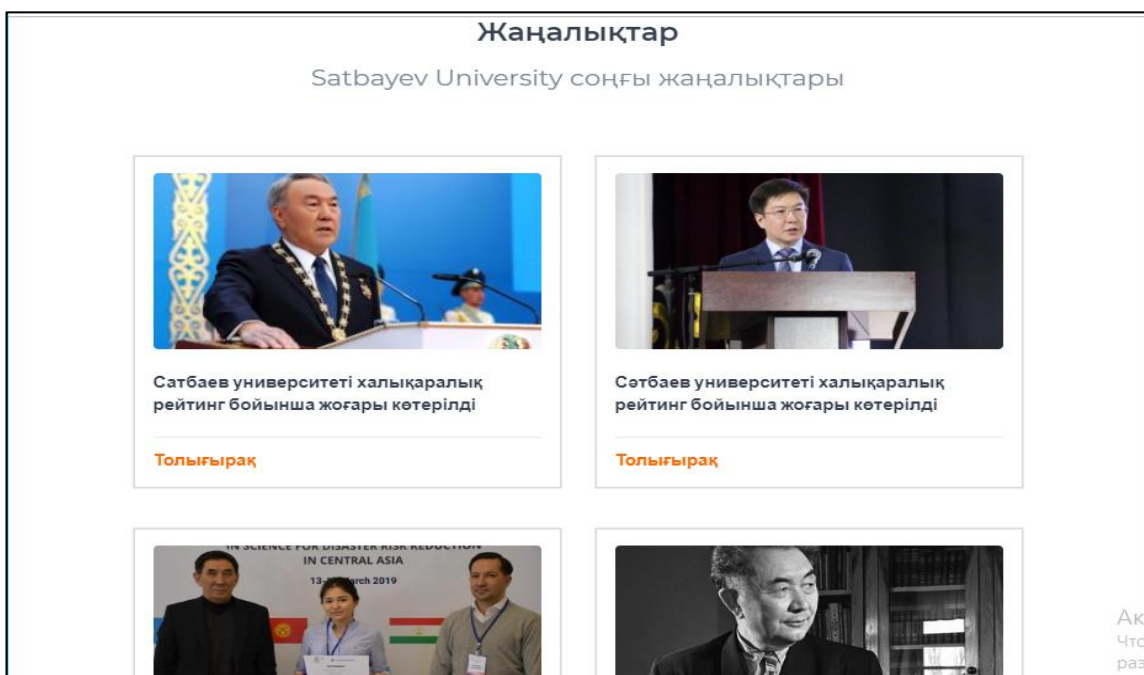
### **4.2 Қолданушының интерфейсі**

Бұл проектіні жасаудағы басты мақсатым веб бетті желіге орналастырып, мобильді қосымшамен REST технологиясы арқылы байланыстыру болды. Мобильді қосымшаны көпшілік қолдана алатындай жүктеп ала алатындай ету болды. Бұл веб сайтты кез келген браузерде Yandex, Mozilla, Internet Explorer, Google Chrome сынды браузерлерде ашуға болады. Веб сайтты хостингке салу арқылы қолданушылар тартып ала алады. Веб сервер ретінде Apache веб серверін таңдап алдым.

Ең алдымен веб бетке кірген кезде университет туралы қысқаша ақпараттарды көре аласыз.

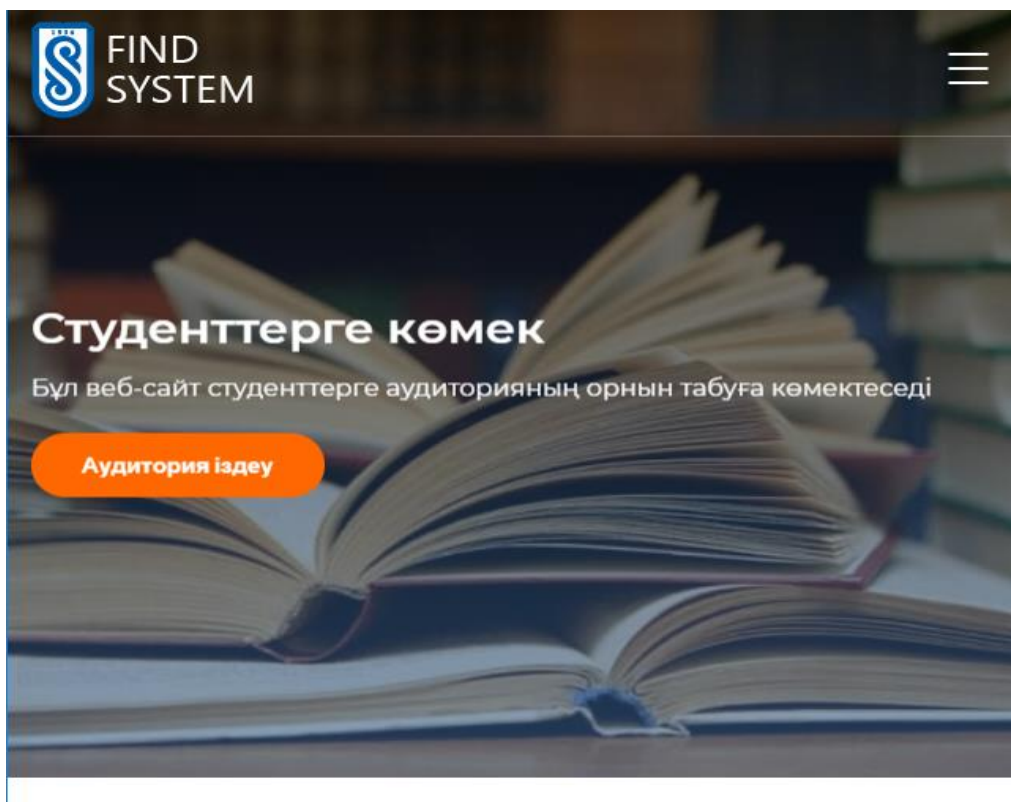


4.2-сурет – Веб-беттің басты беті



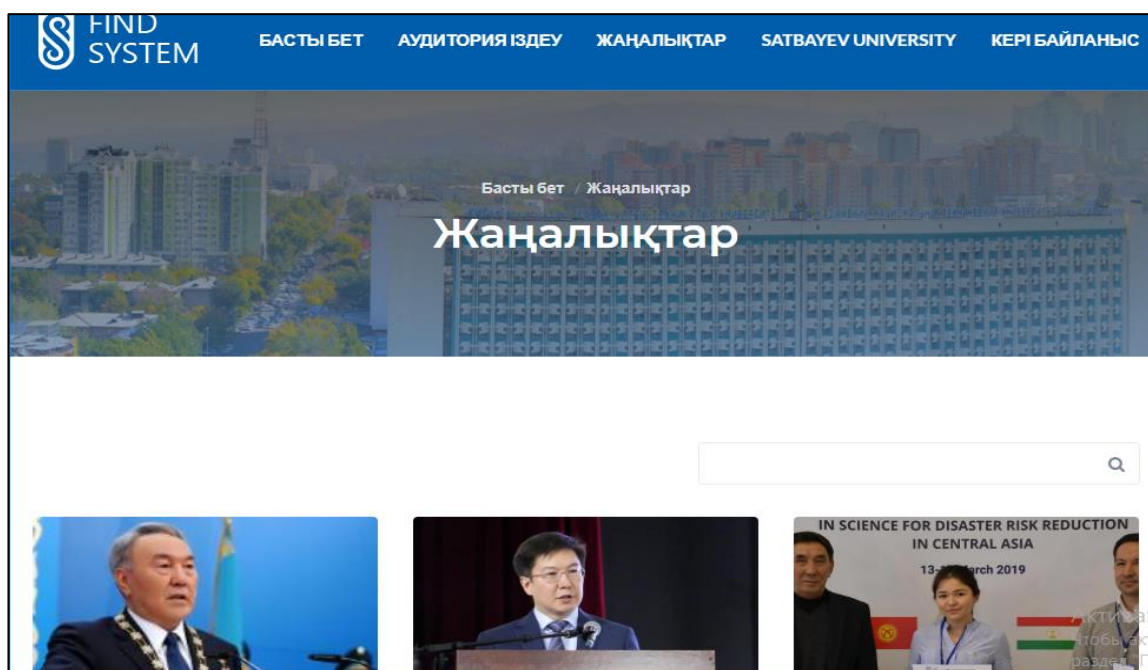
4.2-сурет – Веб-беттің басты беті

Веб сайт адаптивті болып келеді. Ұялы телефоннан немесе басқа құрылғыдан қарағанда да көзге жағымды және ыңғайлы көрінеді. Яғни қандай құрылғы да ашса да веб бетті ыңғайлы қолдана алады. Bootstrap тірегі сайттың дизайнына жауап береді. Қолданушыларға жағымды етіп көрсетеді. Сайттың жоғарғы жағында мәзір орналасқан. Ал баннерді админ беттен қосып, жаңартып отыруға болады. Ал ұялы телефоннан веб бетті ашқанда 4.3 суреттегі қалыпқа енеді.



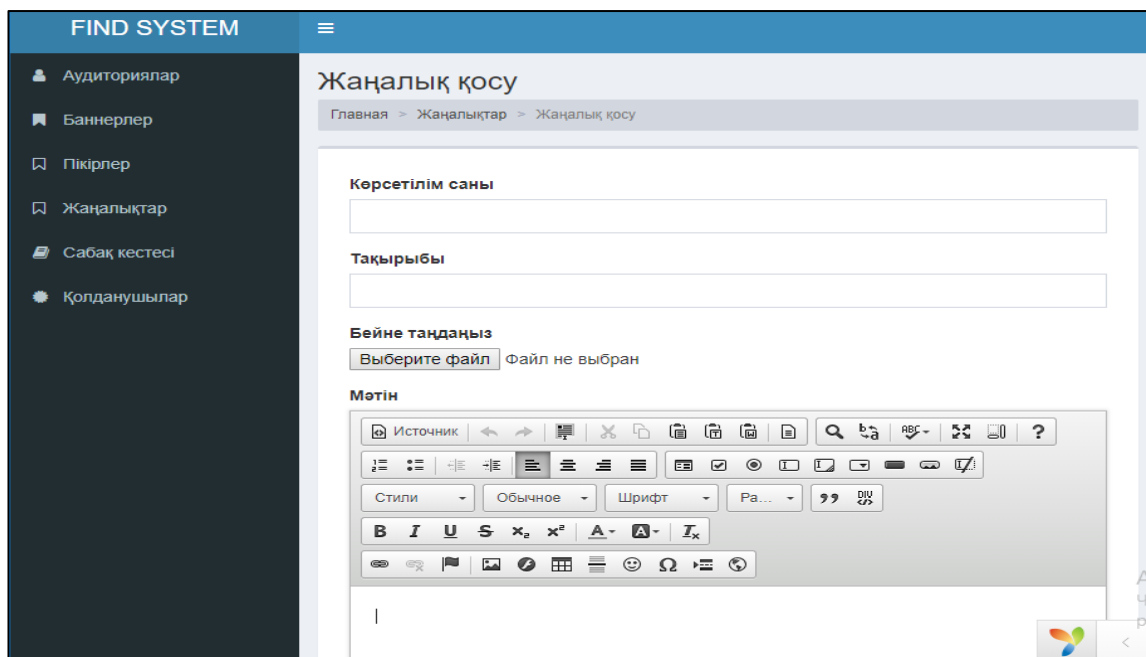
4.3-сурет – Веб-беттің ұялы телефоннан қарағандағы түрі

Веб беттің «Жаңалықтар» бөлімінде университет жайлы соңғы ақпараттарды көруге болады. Өзіңізге керек жаңалықты оқығыңыз келсе сол жаңалыққа өте аласыз. Егер жаңалық туралы пікірлеріңіз болса төменгі жақта пікір қалдыру бөлімінде өзіңіздің email жазып пікір қалдыра аласыз.



4.4-сурет – Жаңалықтар бөлімі

«Жаңалықтар» бөліміндегі жаңалықтарды қосу, өзгерту немесе жою сияқты жұмыстар админ бетінде орындалады. Жаңалық қосу бұл 4.5-суретте көрсетілген.



**4.5-сурет – Админ бетте жаңа жаңалық қосу**

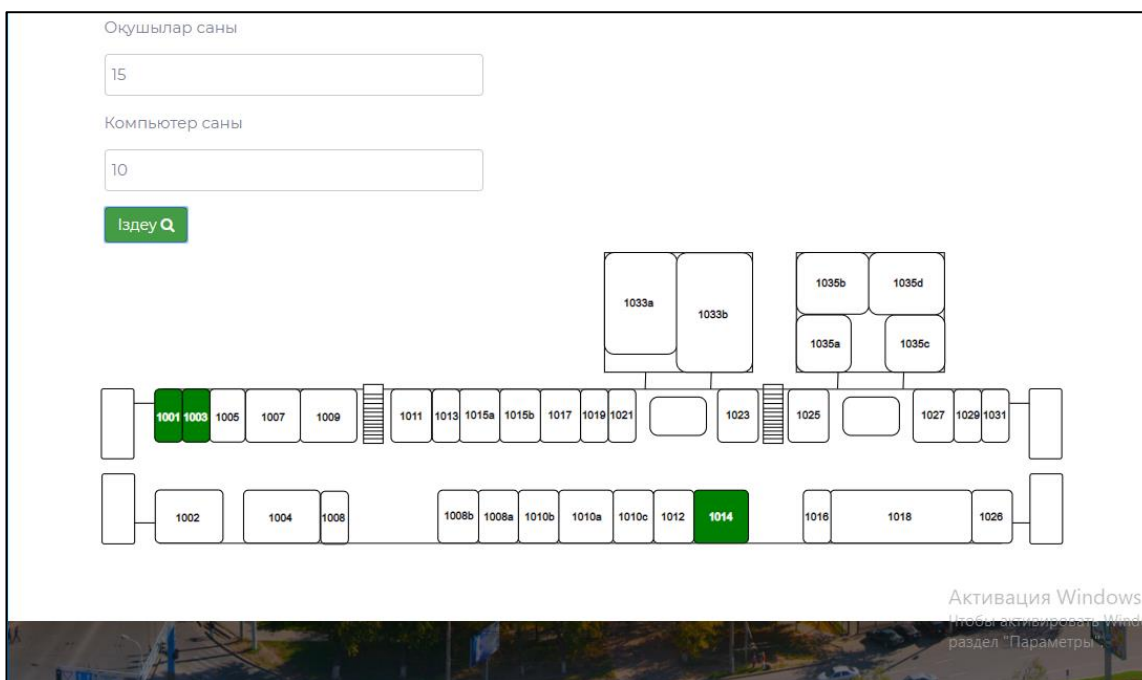
Веб беттің «Аудитория іздеу» мұғалімдерге арналған бөлімінде Аудитория іздеу батырмасын басқан кезде сізге этаж, оқушылар саны және компьютерлер саны деген жолдар шығады. Өзіңізге керек жолдарды толтыру арқылы керек аудиторияны таба аласыз.

Аудитория іздегенде енгізетін ақпараттар

- этаж
- оқушылар саны
- компьютерлер саны

#### 4.6-сурет – Аудитория іздеу бөлімі

Іздеу батырмасын басқан кезде сізге аудиториялар жасыл түспен белгіленіп көрсетіледі.

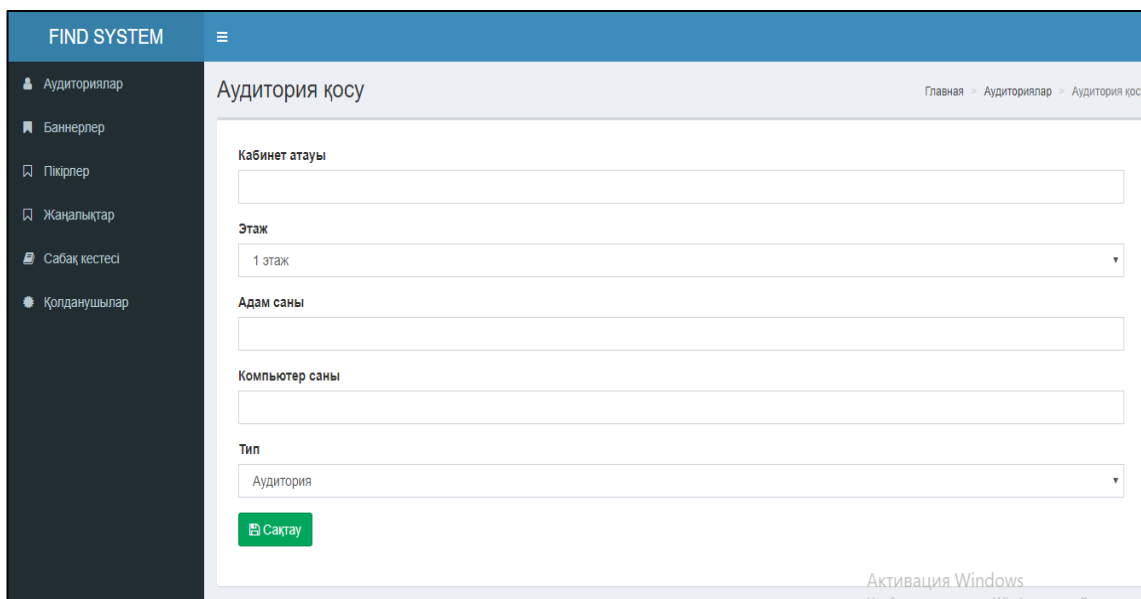


#### 4.7-сурет – Сұрауға байланысты шыққан аудиториялар

Аудиторияларды қосу, жою қызметтері админ бетінде орындалады. Аудитория картасын сызу үшін SVG графикалық редакторын қолданып жасадым.

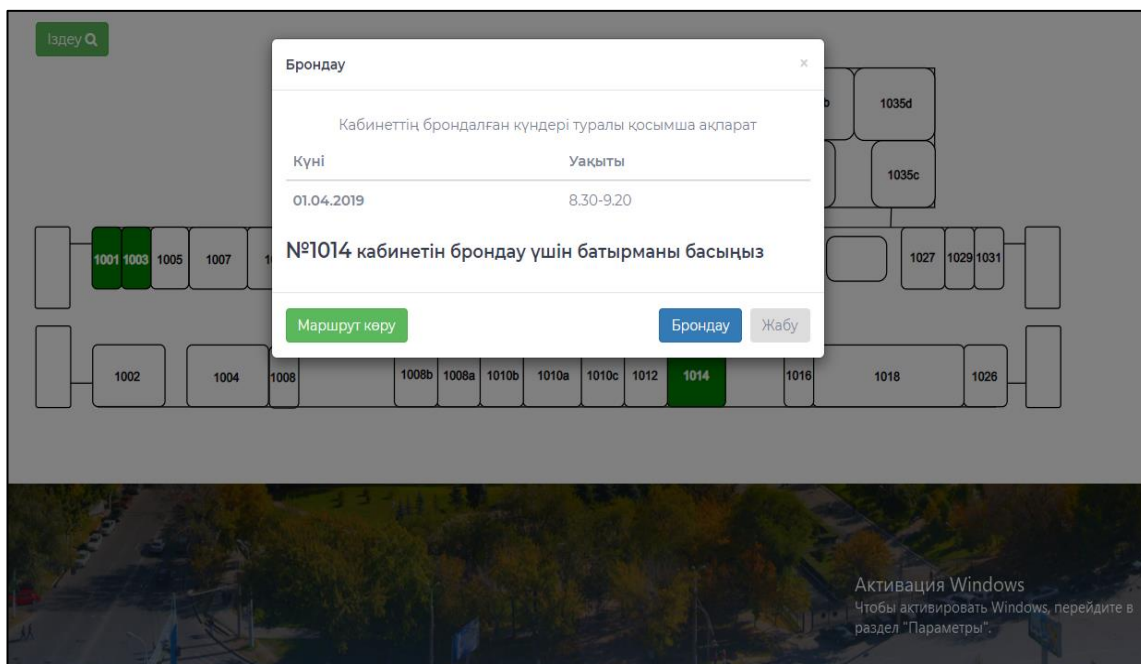


SVG-edit-векторлық графиканың онлайн редакторы. Оны веб-қосымша ретінде (кез келген браузерде) немесе онлайн-сервис ретінде пайдалануға болады. SVG файлдарын және басқа пішімдерді жасау және өңдеу үшін қолданылады. Бағдарлама JavaScript-те жазылған.



**4.8-сурет- Админ бетінен аудитория қосу**

Веб бетке өтіп аудиторияны брондау үшін кабинеттің үстін басып брондау батырмасын басу жеткілікті. Маршрутты көру батырмасын басу арқылы аудиторияға тиімді жолды пайдаланып баруға болады.

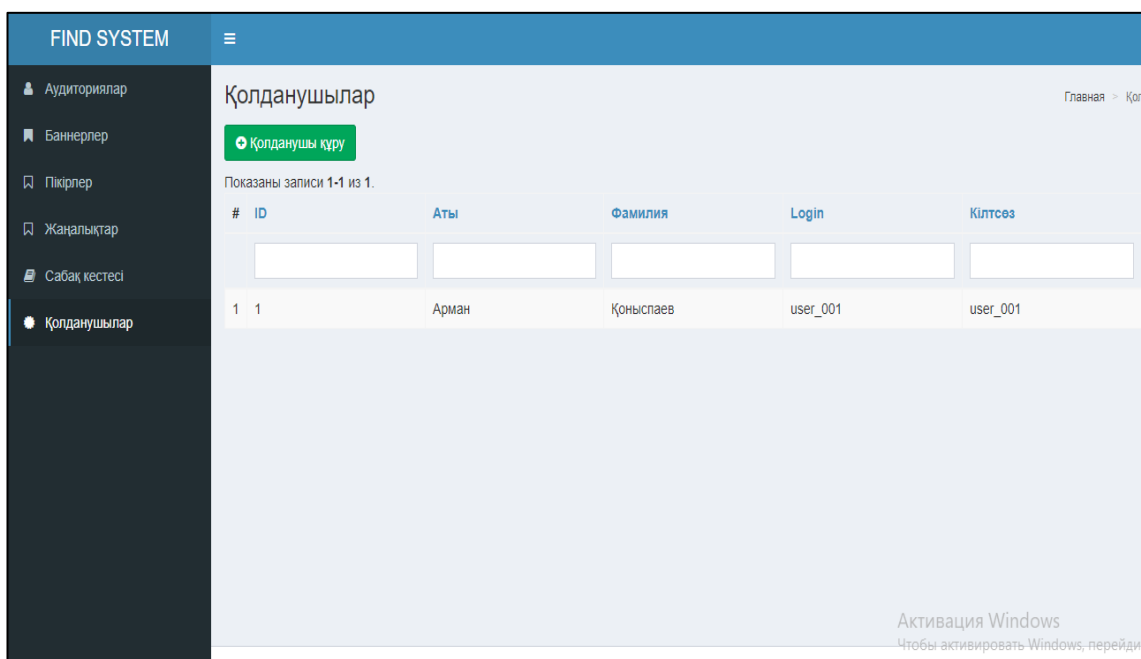


**4.9-сурет – Аудиторияны брондау жүйесі**

Брондау кезінде орындалатын қызметтер:

- күнін таңдау
- уақытын таңдау
- логин мен құпия сөзді енгізу

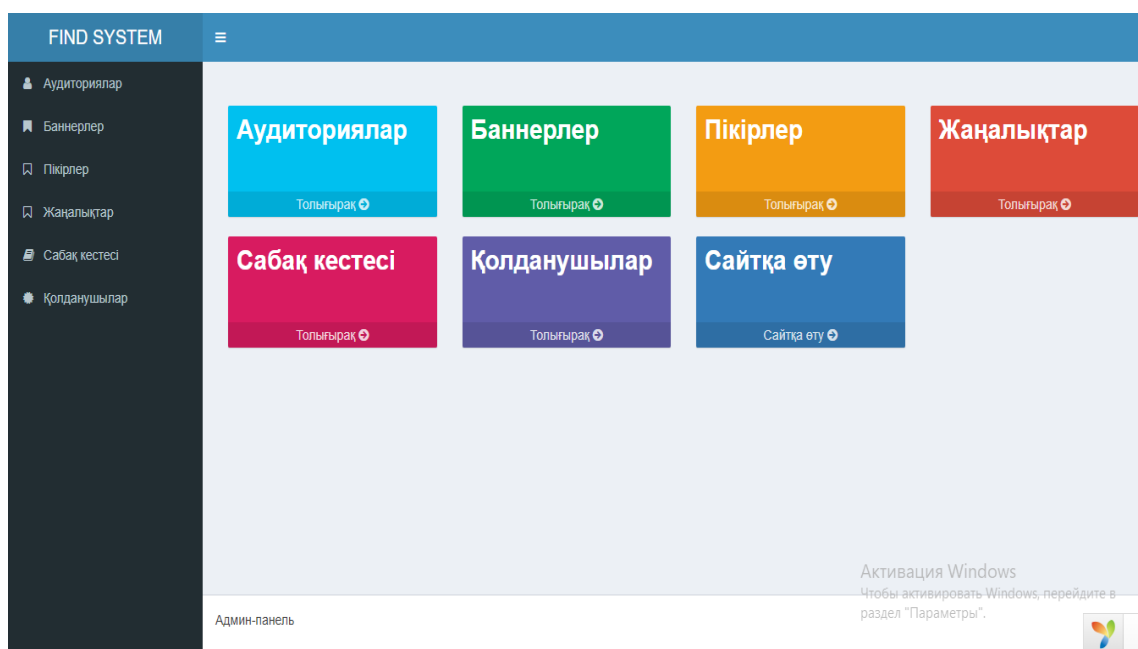
Өзіне керек аудиторияны брондау жүйесінде оқытушының логині мен құпия сөзі керек. Логин және құпия сөз жазу арқылы брондау жүйесі орындалады. Админ беттен оқытушының логин және құпия сөзін беріп кетеміз.



**4.9-сурет – Жаңа қолданушы қосу**

Веб сайттың админ бетінде мынадай бөлімдерге бөлінген:

- аудиториялар
- баннерлер
- пікірлер
- жаңалықтар
- сабақ кестесі
- қолданушылар
- сайтқа өту



**4.10-сурет – Админ беті**

## ҚОРЫТЫНДЫ

Қазіргі заманда әрбір мекеменің өзінің жеке Интернет әлемінде бір өкілі бар. Қазір веб сайт кез келген мәселелер бойынша әлеуметтік қолданушылармен байланысуы мүмкін. Мұндай веб сайттар серверлік бөлікті пайдалана отырып, клиент-сервер технологиясымен жұмыс жасайды. Бұл жобалар кез келген бағдарламалау тілінде жүзеге асырылады.

Бұл проектінi жасаудағы басты мақсатым бұл веб сайтты қолданушыларға барлық қолайлылықты қарастыру. Осы сияқты ұқсас жобалардан айырмашылығын көрсету.

Жобаның мақсаты мобильді қосымшаға да, вет бетке тұрақты және қарапайым API құру болып табылады. Ол API деректер базасымен сенімді байланысады.

Веб бет үлкен аудиторияға арналған. Негізінде университетке қатысы бар адамдарға. Яғни мұғалімдер, студенттер және университет қонақтарына университет жайлы ақпаратты ыңғайлы және тұрақты түрде жеткізу. Бұл веб бет адемі дизайнға ие болуы керек және қолданушыға ыңғайлы интерфейске ие болуы қажет. Жоба вет беттен және мобильді қосымшадан тұрады. Веб бет сервер болсы, Android мобильді қосымшасы клиент болып табылады.

Бұл веб сайтты кез келген браузерден (мысалға, Google Chrome, Internet Explorer, Mozilla Firefox, Yandex) ашып қарауға болады.

Веб бет PHP тілінде Yii2.0 тірегін қолдана отырып жазылды. Yii2.0 тірегі қазіргі кезде қарқынды дамын келе жатқан және сайт жасау үшін бағдарламашылардың тандауы көп түсетін тірек болып табылады.

Android мобильді қосымшасы мен веб-бет арасындағы байланысты Rest технологиясы жүзеге асырады. Restful API деректерді JSON форматында қайтарады.

## ПАЙДАЛАНЫЛҒАН 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 - РНР 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 РНР. – 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 РНР 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. РНР 5: Практика создание Web – сайтов. СПб.: БХВ – Петербург, 2007. – 150 с.
- 16 Лаура Томсон, Люк Веллинг. Разработка Web-приложений на РНР и MySQL. – СПб.: БХВ – Петербург, 2008. – 365 с.

## **А қосымшасы**

«Find System» веб-бетін құруға арналған техниалық тапсырма

### **1 Кіріспе**

Satraev University еліміздің ауқымды оқу орындары болып табылады. Бұл веб-сайт Satraev University –нің новигациялық гиді болып табылады. Веб-сайт php тілінде My Sql базасын қолдану арқылы жасалынады. Веб-сайтты кез келген адам көре алады.

#### **1.1 Мақсаты мен қолданылу аясы**

Университетімізге арналған “Find system” веб-бетін құру студенттерге, оқытушыларға кез келген жерде және кез келген уақытта университетіміз туралы барлық қажетті ақпаратты алуға мүмкіндік береді. Проектінің түпкі мақсаты, университетімізде оқитын студенттерге, оқытушыларға немесе абитуриенттерге аудиторияның орнын табуға және аудитория жайлы ақпарат алуды жеңілдету және жылдамдату. Брондау жүйесі оқытушыларға қалаған аудиториясында сабақ беру мүмкіндігін береді.

Сайтқа мыналар кіреді:

1. Интерактивті карта:
  - Университетіміздің басты ғимараты БОҒ - тың аудиторияларын картадан іздеу.
  - Сәтпаев университетінің оқытушылары аудиторияны брондау мүмкіндігіне ие.
2. Сабақ кестесіне қол жеткізу
  - Аудитория бойынша сабақ кестесін қарау
3. Жаңалықтар:
  - Университет жаңалықтары мен хабарландырулары;

### **2 Сайтқа қойылатын талаптар**

#### **2.1 Функционалдық талаптар**

Сайт төмендегі функциялардың орындалуын қамтамасыз ету керек:

Сайтпен байланысатын пайдаланушыларды топтарға бөлу

- Резидент(студенттер,оқытушылар,ҚазҰТУ-дың қызметкерлері);

- Резиденттер емес
- БОҒ аудиториялары туралы ақпарат алу мүмкіндігі “екі категория үшін”
- Сабақ кестесі, өзгерістер туралы хабарламаларды алу мүмкіндігі «студенттерге» арналған кесте;
- Жаңалықтарды жылдам іздеу

## **2.2 Сенімділікке қойылатын талаптар**

Сайттың сенімді (тұрақты) жұмыс істеуі қажет. вирусқа қарсы осалдығын нөлге дейін азайту керек. Олай болмаған жағдайда, Қолданба оған тағайындалған функцияларды орындай алмайды жарамсыз болады. Осы проблемаларды шешкеннен кейін, бағдарламаны жеңіл қылып жасау қажет, өйткені интернет желісіне қосылу үлкен жылдамдықты қажет етпейтіндей және пайдаланушының трафигін аз мөлшерде жұмсайтындай етіп.

## **3 «Бағдарламаны әзірлеу» кезеңі бойынша жұмыстың мақсаты**

- 1 Жобалау (қағаз бетінде).
  - 1.1 Құрылымды құрастыру (қайда және қалай болатынын шешу)
  - 1.2 Интерфейс үшін түс таңдау
  - 1.3 Бастапқы жобаны мақұлдау
- 2 LMS дерекқорларын (студенттер туралы) интеграциялау туралы мәселені шешу.
  - 2.1 Бұл бастапқы бетті бағдарлама жобасына қосады пайдаланушыларды сұрыптайды. Студенттердің дерекқорын алу қажет.
  - 2.2 LMS әкімшілері дерекқорларды ұсынудан бас тартқан жағдайда, онда пайдаланушыларға арналған басқа сұрыптау жүйесін жасау керек.
  - 2.3. Мәселені шешу: студенттер саны берілген базаны алу үшін оқыту кеңсесіне хабарласу. Сұрыптау кезеңінің соңында пайдаланушыға телефон нөмірін енгізу мүмкіндігі беріледі.
- 3 Қолданбаны тексеріңіз
  - 3.1 Белсенді аумақтарды басқанда, аудитория, кабинет туралы ақпарат ала аламыз.
  - 3.2 Екі жақтың баспалдағынан қажетті аудиторияға дейін маршрут.
  - 3.4 Сабақ кестесіндегі өзгерістер туралы хабарландыру жүйесін жасау.
- 4 Жаңалықтар алуды ұйымдастыру
  - 4.1 Факультеттің бетінен;
  - 4.2 Оқу бағдарламасының бетінен; (таңдау функциясы болуы керек бағдарлама)

5 Жаңалық жүйесіне таңдау қосу Оқу факультеті. Демек, факультеттердің базасы және осы факультеттердегі бағдарламалар. Барлық функциялар бірдей қалады

6 Бағдарламаны түпкілікті тестілеу



**Б қосымшасы**  
(міндетті)

*Бағдарламаның мәтіні*

```
<?php
defined('YII_DEBUG') or define('YII_DEBUG', true);
defined('YII_ENV') or define('YII_ENV', 'dev');

require(__DIR__ . '/../vendor/autoload.php');
require(__DIR__ . '/../vendor/yiisoft/yii2/Yii.php');
require(__DIR__ . '/../common/config/bootstrap.php');
require(__DIR__ . '/../config/bootstrap.php');

$config = yii\helpers\ArrayHelper::merge(
    require(__DIR__ . '/../common/config/main.php'),
    require(__DIR__ . '/../common/config/main-local.php'),
    require(__DIR__ . '/../config/main.php'),
    require(__DIR__ . '/../config/main-local.php')
);

(new yii\web\Application($config))->run();
<?php

/* @var $this \yii\web\View */
/* @var $content string */

use yii\helpers\Html;
use frontend\assets\AppAsset;
use common\widgets\Alert;

AppAsset::register($this);
?>
<?php $this->beginPage() ?>
<!DOCTYPE html>
<html lang="<?= Yii::$app->language ?>">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content
    must come *after* these tags -->
```

## *Б қосымшасының жалғасы*

```
<title>Аудитория табуға арналған веб-сайт</title>

<link rel="shortcut icon" href="/frontend/web/img/favicon.png" type="image/x-
icon" >

<!-- Google font -->
<link
href="https://fonts.googleapis.com/css?family=Lato:700%7CMontserrat:400,600"
rel="stylesheet">

<!-- Bootstrap -->
<link type="text/css" rel="stylesheet" href="/frontend/web/css/bootstrap.min.css"/>

<!-- Font Awesome Icon -->
<link rel="stylesheet" href="/frontend/web/css/font-awesome.min.css">

<!-- Custom stylesheet -->
<link type="text/css" rel="stylesheet" href="/frontend/web/css/style.css"/>
<link type="text/css" rel="stylesheet" href="/frontend/web/css/owl_carousel.css"/>
<link type="text/css" rel="stylesheet" href="/frontend/web/css/aos.css"/>

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media
queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<?= Html::csrfMetaTags() ?>
<title><?= Html::encode($this->title) ?></title>
<?php $this->head() ?>
</head>
<body>
<?php $this->beginBody() ?>

<!-- Header -->
<header id="header" class="not-main" data-aos="fade-down">
<div class="container">

<div class="navbar-header">
<!-- Logo -->
```

## Б қосымшасының жалғасы

```
<div class="navbar-brand">
  <a class="logo" href="/">
    
  </a>
</div>
<!-- /Logo -->

<!-- Mobile toggle -->
<button class="navbar-toggle">
  <span></span>
</button>
<!-- /Mobile toggle -->
</div>

<!-- Navigation -->
<nav id="nav">
  <ul class="main-menu nav navbar-nav navbar-right">
    <li><a href="/">Басты бет</a></li>
    <li><a href="/frontend/web/auditory">Аудитория іздеу</a></li>
    <li><a href="/frontend/web/auditory/student">Студентке</a></li>
    <li><a href="/frontend/web/posts">Жаңалықтар</a></li>
    <li style="position:relative;">
      <a href="#" class="btn_dropdown">Satbayev University</a>
      <div class="dropdown_body">
        <a href="/frontend/web/site/university">Тарихы</a>
        <a href="/frontend/web/schedule">Сабақ кестесі</a>
      </div>
    </li>
    <li><a href="/frontend/web/contacts">Кері байланыс</a></li>
  </ul>
</nav>
<!-- /Navigation -->

</div>
</header>
<!-- /Header -->

<?= $content ?>

<!-- Contact CTA -->
<div id="contact-cta" class="section">
```

## Б қосымшасының жалғасы

```
<!-- Background Image -->
<div class="bg-image bg-parallax overlay" style="background-
image:url(/frontend/web/img/bg.jpg)"></div>
<!-- Background Image -->

<!-- container -->
<div class="container">

  <!-- row -->
  <div class="row" data-aos="zoom-in">

    <div class="col-md-8 col-md-offset-2 text-center">
      <h2 class="white-text">Кері байланыс</h2>
      <p class="lead white-text">Сұрақтарыңыз болса бізге хабарлама
жіберіңіз!</p>
      <a class="main-button icon-button"
href="/frontend/web/contacts">Хабарлама жазуға өту</a>
    </div>

  </div>
<!-- /row -->

</div>
<!-- /container -->

</div>
<!-- /Contact CTA -->

<!-- Footer -->
<footer id="footer" class="section">

  <!-- container -->
  <div class="container">

    <!-- row -->
    <div class="row">

      <!-- footer logo -->
      <div class="col-md-4">
        <div class="footer-logo">
          <a class="logo" href="/">
```

## Б қосымшасының жалғасы

```
        
      </a>
    </div>
  </div>
  <!-- footer logo -->

  <!-- footer nav -->
  <div class="col-md-8">
    <ul class="footer-nav">
      <li><a href="/">Басты бет</a></li>
      <li><a href="/frontend/web/auditory">Аудитория іздеу</a></li>
      <li><a href="/frontend/web/auditory/student">Студентке</a></li>
      <li><a href="/frontend/web/posts">Жаңалықтар</a></li>
      <li><a
                href="/frontend/web/site/university">Satbayev
University</a></li>
      <li><a href="/frontend/web/contacts">Кері байланыс</a></li>
    </ul>
  </div>
  <!-- /footer nav -->

</div>
<!-- /row -->

<!-- row -->
<div id="bottom-footer" class="row">

  <!-- social -->
  <div class="col-md-4 col-md-push-8">
    <ul class="footer-social">
      <li><a
        href="https://www.instagram.com/satbayev_university/"
class="instagram" target="_blank"><i class="fa fa-instagram"></i></a></li>
      <li><a
        href="https://vk.com/satbayevuniversity"
class="linkedin"
target="_blank"><i class="fa fa-vk"></i></a></li>
      <li><a
        href="https://www.youtube.com/user/TheKazntu/videos"
class="youtube" target="_blank"><i class="fa fa-youtube"></i></a></li>
      <li><a
        href="https://www.facebook.com/satbayevuniversity/"
class="facebook" target="_blank"><i class="fa fa-facebook"></i></a></li>
    </ul>
  </div>
  <!-- /social -->
  <!-- copyright -->
  <div class="col-md-8 col-md-pull-4">
```

```
<div class="footer-copyright">
  <span style="color: #ffffff">© 2019 Satbayev University</span>
</div>
</div>
<!-- /copyright -->

</div>
<!-- row -->

</div>
<!-- /container -->
</footer>
<!-- /Footer -->
<!-- preloader -->
<div id='preloader'><div class='preloader'></div></div>
<!-- /preloader -->

<!--<scripttype="text/javascript"
src="/frontend/web/js/sweet_alert.min.js"></script>-->
<?php $this->endBody() ?>
<!-- jQuery Plugins-->
<script type="text/javascript" src="/frontend/web/js/jquery.min.js"></script>
<script type="text/javascript" src="/frontend/web/js/bootstrap.min.js"></script>
<script type="text/javascript" src="/frontend/web/js/owl_carousel.js"></script>
<script type="text/javascript" src="/frontend/web/js/aos.js"></script>
<script type="text/javascript" src="/frontend/web/js/main.js"></script>

<script
src="https://cdnjs.cloudflare.com/ajax/libs/prefixfree/1.0.7/prefixfree.min.js"></scrip
t>
</body>
</html>
<?php $this->endPage() ?>
<script>
  var date, time, time_text, id;

  let token = $('meta[name=csrf-token]').attr("content");
  // Аудитория этажын ауыстыру
  $('#auditory_floor').on('change', function () {

    $('[data-floor]').hide()
    $('[data-floor='+this.value+']').show()})
```

```
// Басқа типті аудитория этажын ауыстыру
$('#other_floor').on('change', function () {
    $('[data-floor]').hide()
    $('[data-floor='+this.value+']').show()
})

// Аудитория формасы
$('#auditory_form_button').on('click', function (e) {
    e.preventDefault();
    var floor = $('#auditory_floor').val();
    var capacity = $('#capacity').val();
    var computer_quantity = $('#computer_quantity').val();
    var auditory_type = $('#auditory_type').val();
    if (capacity != "") {
        if (computer_quantity == "") {
            computer_quantity = 'pustoi'
        }
        $.ajax({
            dataType: "json",
            type: "POST",
            url: "/frontend/web/auditory/quantity",
            data: {
                "_csrf-frontend": token,
                floor: floor,
                auditory_type: auditory_type,
                capacity: capacity,
                computer_quantity: computer_quantity
            },
            success: function (data) {
                $('#rect').removeClass('active inactive');
                $('#text').removeClass('active inactive');
                if(data.auditory_names.length != 0) {

                    $.each( data.auditory_names, function( index, value ){

                        $('#cabinet_'+value).addClass('active')
                        $('#text_'+value).addClass('active')
                    });
                } else if (data.auditory_names.length == 0) {
                    SendSwal("Табылмады", "error", null)
                }
            }
        })
        // SendSwal(data.message, data.type, null)
    }
})
```

```
    },
    error: function (e) {
        console.log(e.responseText);
    }
});
} else {
    SendSwal("Адам санын толтырыңыз!", "error", null)
}
})
```

// Басқа типті аудитория формасы

```
$('#other_form_button').on('click', function (e) {
    e.preventDefault();
    var other_floor = $('#other_floor').val()
    var other_type = $('#other_type').val()
    if (other_type == 'Дәретхана') {
        $('rect').removeClass('active inactive');
        $('text').removeClass('active inactive');
        $('toilet').addClass('active pointer_none')
    } else {
        $.ajax({
            dataType: "json",
            type: "POST",
            url: "/frontend/web/auditory/other-types",
            data: {
                "_csrf-frontend": token,
                floor: other_floor,
                auditory_type: other_type,
            },
            success: function (data) {
                $('rect').removeClass('active inactive');
                $('text').removeClass('active inactive');
                if(data.auditory_names.length != 0) {
                    $.each( data.auditory_names, function( index, value ){

                        $('#cabinet_'+value).addClass('active pointer_none')
                        $('#text_'+value).addClass('active pointer_none')
                    });
                } else if (data.auditory_names.length == 0) {
                    SendSwal("Табылмады", "error", null)
                }
            }
        })
        // SendSwal(data.message, data.type, null)
    }
})
```



```
    },
    error: function (e) {
        console.log(e.responseText); } }); } })

// Кабинетті басқан кезде
$('body').on('click', 'rect', function () {
    var cabinetNumber = $(this).attr('id');
    var substNumber = cabinetNumber.substring(8);
    // ВНИМАНИЕ КОСТЫЛЬ
    // console.log('#1 '+substNumber)
    $.ajax({
        dataType: "json",
        type: "POST",
        url: "/frontend/web/booking/table",
        data: {
            "_csrf-frontend": token,
            auditory_id: substNumber,
        },
        success: function (data) {
            $('#tbody').html(data.html)
            $('#booking_modal').modal();
            // SendSwal(data.message, data.type, null)
        },
        error: function (e) {
            console.log(e.responseText);
        }
    });

    $('.view_number').html(substNumber);
    $('.marshrut').hide();

    $('.marshrut_btn').attr('data-id', substNumber);
    // $('#booking_btn').attr('data-id', substNumber);
    $('#booking_btn').data("id", substNumber);
    $('#booking_number').val(cabinetNumber);
});

// КҮН МЕН УАҚЫТТЫ ТАҢДАҒАНДА
$('#booking_btn2').on('click', function (e) {
    e.preventDefault();
    date = $('#booking_form2_date').val()
    time = $('#booking_form2_time').val()
```

*Б қосымшасының жалғасы*

```
time_text = $("#booking_form2_time :selected").text();  
id = $('#booking_btn2').attr('data-id');  
})
```

```
// Кіру батырмасын басқанда  
var login_form = $('#login_form');  
login_form.on('submit',function(e){  
    e.preventDefault();  
    var login = $('#login').val();  
    var password = $('#password').val();  
    $.ajax({  
        dataType: "json",  
        type: "POST",  
        url: "/frontend/web/booking/login",  
        data: {  
            "_csrf-frontend": token,  
            login: login,  
            date: date,  
            password: password,  
            time: time,  
            time_text: time_text,  
            id: id  
        },  
        success: function (data) {  
            SendSwal(data.message, data.type, null)  
            if (data.type == 'error') {  
                $('rect').removeClass('active inactive');  
                $('text').removeClass('active inactive');  
                $('#cabinet_'+id).addClass('inactive')  
                $('#text_'+id).addClass('active')  
            }  
            $('#login_modal').modal('toggle');  
        },  
        error: function (e) {  
            console.log(e.responseText);  
        }  
    });  
})
```

```
function SendSwal(message, type, link) {  
    swal({  
        title: message,
```

*Б қосымшасының жалғасы*

```
    type: type,  
    showConfirmButton: true  
  });  
  // if (link != null) {  
  //   $('#' + link).trigger('click');  
  // }  
}  
  
</script>
```

Қ.И. Сәтбаев атындағы Қазақ Ұлттық техникалық зерттеу университеті

Мамандығы 5В060200 – Ақпараттану

Студент Файзулла Әйгерім

Тақырыбы: «Find System» проектісінің веб-бетін құру

### ҒЫЛЫМИ ЖЕТЕКШІНІҢ СЫН ПІКІРІ

Бұл дипломдық жобада университетімізге арналған веб-бет қарастырылған. Жобаның түпкі мақсаты Satpaev University оқытушылары мен осында білім алып жатқан студенттердің оқу жүйесін автоматтандыру болып табылады. Өзіне қажетті аудиторияны іздеп, сонымен қатар аудитория жайлы ақпаратқа қол жеткізе алады. Брондау жүйесі оқытушыларға қалаған аудиториясында сабақ беру мүмкіндігін береді. Жаңалықтар бөлімі университетіміздегі соңғы жаңалықтарымен хабардар етіп отырады. Студент басқаларды қызықтыра алатындай дипломдық жоба жасаған.

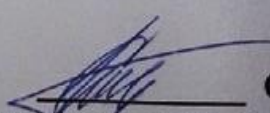
Дипломдық жұмыстың бірінші бөлімінде тақырыпқа байланысты веб-бет жасаудағы өзектіліктерге талдау жасалынды. Екінші бөлімі жобалау бөлімі, модельдеу арқылы диаграммаларды қолдана отырып дипломдық жобаны жобалап алды. Үшінші бөлім жобада қолданылатын программалық қамтамалар туралы. Ал соңғы төртінші бөлімде веб-бетінің интерфейсі жайлы, бағдарламалық құрылымы, ерекшеліктерін қарастырған.

Студент диплом жұмысын жазу процесі кезінде материалды зерттеу талдау және жүйелеу әдістерін, безендіру тәсілдерін дұрыс пайдаланған, диплом жазушы «Тапсырма» бойынша барлық қойылған талаптарды толық орындаған.

Қорытындылай келе студент өз мүмкіндіктерін кеңінен пайдаланып, дипломдық жобаны жақсы жасап шыққан.

Жоба жетекшісі ретінде студент Файзулла Әйгерім Асқарқызы «Find System веб-қосымшасы» атты тақырыптағы дипломдық жұмысын толық аяқтап өткізді.

ҒЫЛЫМИ ЖЕТЕКШІ  
«Программалық инженерия» кафедрасының  
сениор-лекторы «17» май 2019 жыл

  
С.Б. Қалдыбеков