

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ  
«АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ»  
коммерциялық емес акционерлік қоғамы  
IT-инжиниринг кафедрасы

**ҚОРҒАУҒА ЖІБЕРІЛДІ**

Кафедра меңгерушісі

PhD, доцент

\_\_\_\_\_ Т.С. Картбаев  
« \_\_\_\_ » \_\_\_\_\_ 2019 ж.

**ДИПЛОМДЫҚ ЖОБА**

Тақырыбы: Университет түлектерінің бірлестігіне арналған веб-бағытталған жүйе құру

Мамандығы: 5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»

Орындаған: Қалидоллина Г.Т.

Тобы: ВТк-15-1

Ғылыми жетекші: PhD, доцент Т.С. Картбаев

Кеңесшілер:

Экономикалық бөлім: э.ғ.к., профессор \_\_\_\_\_ Ж.Г. Аренбаева  
« 10 » 05 2019 ж.

Өміртіршілік қауіпсіздігі: т.ғ.д., аға оқытушы \_\_\_\_\_ Ш.Ш. Бекбасаров  
« 14 » 05 2019 ж.

Есептеу техникасын қолдану: аға оқытушы \_\_\_\_\_ Ж.С. Айткулов  
« 13 » 05 2019 ж.

Норма бақылаушы: аға оқытушы \_\_\_\_\_ К. Мукапил  
« 17 » 05 2019 ж.

Сын-пікір беруші: PhD, асс. проф. \_\_\_\_\_ Н.К. Мукажанов  
« 20 » 05 2019 ж.

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ  
«АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ»  
коммерциялық емес акционерлік қоғамы

Басқару жүйелері және ақпараттық технологиялар институты

IT-инжиниринг кафедрасы

Мамандығы 5B070400 – «Есептеу техникасы және  
бағдарламалық қамтамасыз ету»

Дипломдық жобаны орындауға берілген  
**ТАПСЫРМА**

Білім алушы Қалидоллина Гүлмаржан Талғатқызы

Жобаның тақырыбы: Университет түлектерінің бірлестігіне арналған  
веб-бағытталған жүйе құру

2018 жылғы «26» қазан № 124 университет бұйрығымен бекітілген.

Аяқталған жобаны тапсыру мерзімі: «24» мамыр 2019 ж.

Дипломдық жобаның бастапқы мәліметтері (зерттеу (жоба) нәтижелерінің талап етілген параметрлері мен объектінің бастапқы мәліметтері): Ұсынылып отырған дипломдық жобада Университет түлектерінің бірлестігіне арналған веб-бағытталған жүйе құру. Жобаны орындау барысында PHP скрипт тілінің құрамдас құрылымына негізделген Yii2 жоғары тиімді фреймворкті және MySQL ДҚБЖ қолданамын.

Дипломдық жобада қарастырылған мәселелер тізімі немесе дипломдық жобаның қысқаша мазмұны:

- талдау бөлімі;
- жобалау бөлімі;
- жүзеге асыру және тестілеу бөлімі;
- экономикалық бөлім;
- өміртіршілік қауіпсіздігі;
- А қосымшасы. Техникалық тапсырма;
- Ә қосымшасы. Программа листингі;
- Б қосымшасы. Ендіру актісі.

Графикалық материалдар тізімі (міндетті сызбалар дәл көрсетілуі тиіс):  
9 кесте, 67 сурет ұсынылған.

Ұсынылатын негізгі әдебиеттер:

1 Бардзелл, Джеффри Macromedia Dreamweaver MX 2004 с ASP, ColdFusion и PHP. Из первых рук (+ CD-ROM) / Джеффри Бардзелл. - М.: Эком, 2017. - 560 с.

2 Дронов, В.А. PHP 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов / В.А. Дронов. - М.: БХВ-Петербург, 2012. - 418 с.

3 Колисниченко, Денис Профессиональное программирование на PHP (+CD-ROM) / Денис Колисниченко. - М.: БХВ-Петербург, 2015. - 416 с.

4 Кузнецов, М. Объектно-ориентированное программирование на PHP / М. Кузнецов, И. Симдянов. - М.: БХВ-Петербург, 2012. - 608 с.

5 Гультияев, А. К. Уроки Web-мастера. Технология. Дизайн. Инструменты / А.К. Гультияев, В.А. Машин. - М.: Корона-Принт, 2016. - 448 с.

Дипломдық жобаның бөлімдеріне қатысты белгіленген кеңес берушілер

Бөлімдер	Кеңесшілер	Мерзімі	Қолы
Экономикалық бөлім	Аренбаева Ж.Г.	04.03.2019 - 10.05.2019	
Өміртіршілік қауіпсіздігі	Бекбасаров Ш.Ш.	04.03.2019 - 14.05.2019	
Программалық қамтама	Айтқулов Ж.С.	04.03.2019 - 02.05.2019	
Норма бақылау	Мукапил К.	04.04.2019 - 10.05.2019	

Дипломдық жобаны дайындау  
КЕСТЕСІ

Бөлімдер атауы, қарастырылатын мәселелер тізімі	Ғылыми жетекшіге ұсыну мерзімдері	Ескерту
Талдау бөлімі	15.11.2018	
Жобалау бөлімі	23.01.2019	
Жүзеге асыру және тестілеу бөлімі	04.02.2019	

Тапсырманың берілген күні « 3 » 09 2018 ж.

Кафедра меңгерушісі Т.С. Картбаев

Жобаның ғылыми жетекшісі Т.С. Картбаев

Тапсырманы орындауға алған білім алушы Г.Т. Қалидоллина

## Аңдатпа

Бұл дипломдық жоба Университет түлектерінің бүрлестігіне арналған веб-бағытталған жүйені жобалауға және жүзеге асыруға арналған.

Дипломдық жобаның негізгі міндеті студенттердің білімі туралы анықтама алуды автоматтандыру және ақпаратқа жылдам қол жеткізу үшін сенімді әрі сапалы жүйені құру болып табылады. Жұмыс барысында қарастырылатын құралға қойылатын талаптар және шешу әдістемесі мен іске асыру құралы қалыптастырылды. Аталған талаптарға сүйене отырып, қазіргі заманғы, тұрақты, масштабталатын веб-бағдарланған жүйені құруға мүмкіндік беретін іске асыру құралы мен әдісі таңдалды.

Жүйе PHP скрипт тілінің құрамдас құрылымына негізделген Yii2 жоғары тиімді фреймворк, сондай-ақ MySQL ДҚБЖ көмегімен жүзеге асырылды.

Әзірленген жүйеде админге өтініш қалдыру арқылы білім туралы анықтаманы онлайн алуға және түлектерді хабарландыру тақтасы арқылы кездесуді ұйымдастыруға болады.

## **Аннотация**

Данный дипломный проект посвящен проектированию и реализации веб-ориентированной системы для ассоциации выпускников университета.

Основной задачей дипломного проекта является автоматизирование получения справок об образовании студентов и создание надежной и качественной системы для быстрого доступа к информации.. В процессе работы были сформированы требования к рассматриваемому инструменту и методика решения и инструмент реализации. Исходя из упомянутых требований был выбран инструмент и метод реализации, которые позволили создать современную, устойчивую, масштабируемую веб-ориентированную систему.

Система реализована с помощью высокоэффективного фреймворка Yii2 основанный на компонентной структуре скриптового языка PHP, а также СУБД MySQL.

В разработанной системе можно оставить заявку администратору и онлайн получить справку об образовании и организовать встречу выпускников посредством доски объявлений.

## **Annotation**

Current thesis work is dedicated to the design and implementation of a web-based system for the association of university graduates.

The main objective of the diploma project is the automation of obtaining certificates of education of students and the creation of a reliable and high-quality system for quick access to information. Furthermore, requirements to the opted tool and a techniques were provided as well as the implementation tool. On the basis of these requirements reliable tools and the method of implementation were selected, which helped to create the modern, robust, scalable web-oriented system.

The system is implemented by use of highly efficient Yii2 framework based on the component structure of the PHP scripting language and MySQL for database.

In the developed system, one can create a request to the administrator and get an online certificate education and also organize graduates gatherings through the bulletin board.

## Мазмұны

Кіріспе	8
1 Пәндік аймақты талдау	9
1.1 Пәндік аймақтың сипаттамасы	9
1.2 Жобалаудың мақсаты мен міндеттерін анықтау	11
1.3 Жобалық шешімдерді талдау және негіздеу	12
1.4 Веб-бағытталған жүйе құрылымы	15
2 Бағдарламалық қамтамасыз етуді жобалау	26
2.1 Бағдарламаны әзірлеу инструменттері	26
2.2 Бағдарламаны әзірлеу құралдары	35
3 Бағдарламалық интерфейс	40
4 Экономикалық бөлім	60
4.1 Экономикалық бөлімде шешілетін мақсаттар мен міндеттер	60
4.2 Қолданбалы бағдарламаны әзірлеудің еңбек сыйымдылығын есептеу	60
4.3 ҚБ әзірлеуге жұмсалатын шығындарды есептеу	60
4.4 Еңбек шығындарын есептеу	62
4.5 Негізгі қорлардың амортизациясын есептеу	64
4.6 ҚБ ықтимал (шарттық) бағасын анықтау	66
4.7 Экономикалық бөлімі бойынша қорытынды	67
5 Өміртіршілік қауіпсіздігі бөлімі	68
5.1 Жарықтандыруды есептеу	68
5.2 Жасанды жарықтандыруды тексеру	71
5.3 Өміртіршілік қауіпсіздігі бөлімі бойынша қорытынды	75
Қорытынды	76
Әдебиеттер тізімі	76
А қосымшасы. Техникалық тапсырма	
Ә қосымшасы. Программа листингі	
Б қосымшасы. Ендіру актісі	

## Кіріспе

Веб-бағытталған жүйе құру интернет ресурстарын әзірлеудің маңызды технологияларының бірі. Жақсы ақпараттық жүйе өзіне барлық пайдалы деректерді жинай отырып, тәуліктің кез келген уақытында қарқынды жұмыс жасайды және коммерциялық фирманың немесе білім беру мекемесінің ең жақсы визит карточкасы болып табылады.

Бүгінгі күні интернетте саяхаттай отырып, тез жүктелетін, тартымды, бөлімдер бойынша нақты шектелген, ақпарат пен навигацияның ыңғайлы функциялары бар өте жақсы орындалған сайттарды табуға болады. Қолданушы мұндай сайттарда біраз уақыт бойы кідіріп, жаңа әрі қызықты ақпарат ала алады. Бұл дипломдық жоба әзірлеу барысында ескеретін маңызды мәселе.

Дипломдық жобаның негізгі міндеттері:

- қолданыстағы жүйелерді зерттеу, талдау және сараптау;
- бағдарлама құру үшін нақты құрал-жабдықтарды таңдау;
- бағдарлама құру барысындағы экономикалық тиімділіктерді зерттеу;
- бағдарлама құру кезіндегі тіршілікке қолайлы ортаны зерттеу;
- бағдарламаны қолданысқа енгізу.

Университет түлектерінің бірлестігіне арналған веб-бағытталған жүйені құру өзектілігі – Университет пен оның түлектері арасындағы өзара тиімді қарым-қатынасқа қол жеткізу, соның ішінде университет тарапынан бітіруші түлектердің кәсіби дамуының барлық кезеңдерінде қолдау көрсетуі және соған орай бітірушілердің университетті дамытуға, ғылыми және білім беру жобаларына ат салысуы болып табылады.

Сайт әзірлеу – бұл күрделі объектпен жасалатын үлкен жұмыс және онымен жұмыс жасау үшін біз оның мақсаттары мен міндеттерін, оның негізгі бағыттарын нақты түсінуге тиіспіз. Қолданушы сайтты құру кезінде қандай технологияларды пайдаланатыны емес, тек жұмыс тиімділігі ғана қызықтырады.

Жобаның мақсаты: түлектерді тарту, ақпаратқа жылдам қол жеткізу үшін сенімді әрі сапалы жүйені құру. Жүйені құруға арналған бағдарламалық қамтамасыз етуді таңдау өте кең және бір шешім арзан әрі жарамсыз, екіншісі қымбат және әмбебап деп айтуға да болмайды, яғни әр түрлі нұсқаларды қарастырып бір оңтайлы шешімге келу керек. Осы шешімдер болашақта сұранысқа ие болуы міндетті, сондықтан басты мақсат – университет түлектерінің тілек-талаптарын ескере отырып, ыңғайлы әрі сенімді ақпараттық, бағытталған жүйе жасау.

Дипломдық жоба келесі бөлімдерден тұрады: пәндік аймақты талдау, бағдарламалық қамтамасыз етуді жобалау, қолданбалы бағдарлама интерфейсі, экономикалық негіздеме, өміртіршілік қауіпсіздігі негіздері және өнім мен қызметтерді қолданысқа енгізу.

Нәтижесінде жоба бойынша қорытынды жасалып, оны объектіге енгізу жолдары және оны одан әрі жетілдіру бағыттары анықталды.



## 1 Пәндік аймақты талдау

### 1.1 Пәндік аймақтың сипаттамасы

Пәндік аймақты талдау – бұл жүйелік талдау кезеңінің бірінші қадамы, одан кейін бағдарламалық жүйені әзірлеу басталады. Әзірлеушілер үйрену керек:

- тапсырыс берушіні толықтай түсіну;
- олардың қызметінің мақсатын анықтау;
- олар шешетін міндеттердің жиынтығын анықтау;
- осы есептерді шешу кезінде мәндер жиынтығын анықтау.

Кез келген ДҚ жобалаудың бірінші кезеңі – концептуалды сұлба (ақпараттық құрылым) құрумен аяқталатын пәндік аймақты талдау болып табылады.

Бұл кезеңде пайдаланушылардың сұраулары талданып, жобаланатын ДҚ мазмұнын анықтайтын ақпараттық объектілер мен олардың сипаттамалары таңдалады. Өткізілген талдау негізінде пәндік аймақ құрылады.

Пәндік аймақты талдау үш кезеңге бөлінеді:

- тұжырымдамалық талаптар мен ақпараттық қажеттіліктерді талдау;
- ақпараттық нысандарды және олардың арасындағы байланыстарды анықтау;
- пән аймағының концептуалды моделін құру және ДҚ тұжырымдамалық сұлбасын жобалау.

Пәндік аймақты талдаудың бірінші фазасы кезінде әзірлеуші осы ДҚ-ның болашақ пайдаланушыларының сұраныстарының тізімін алады.

Пәндік аймақты талдаудың екінші фазасы ақпараттық объектілерді таңдаудан, әрбір объект үшін қажетті қасиеттерді бекітуден, объектілер арасындағы байланыстарды анықтаудан, ақпараттық объектілерге салынатын шектеулерді анықтаудан, олардың арасындағы байланыс типтерінен, ақпараттық объектілердің сипаттамасынан тұрады.

Пәндік аймақты талдаудың қорытынды фазасы оның ақпараттық құрылымын немесе тұжырымдамалық моделін жобалаудан тұрады. Тұжырымдамалық модель деректерді талдау нәтижесінде анықталатын объектілерді және олардың өзара байланысын сипаттауды қамтиды. Концептуалды модель жүйені пайдаланушылардың ақпараттық мүдделерін ескере отырып, пәндік аймақты құрылымдау үшін қолданылады. Ол пайдаланушының пәндік аймаққа көзқарасы болып табылады және тұрақты, яғни өзгеріссіз болуы керек. Оны тек ұзартуға және қосымша деректерді қосуға болады.

Басқаша айтқанда, пәндік аймақты моделдеу жүйенің пәндік аймақ тұрғысынан шешуі тиіс тапсырманы түсінуге көмектеседі. Осы міндетті шешудің кіріктірілген жүйелік әдістері арқылы біз талдау, жобалау және енгізу үдерістерімен айналысамыз.

Бұл дипломдық жұмыста мен зерттегелі отырған пәндік аймақ – «Жоғары оқу орны түлектерінің бірлестігі». Ең алдымен, бұл тақырыптың өзектілігі және Университет өміріндегі маңызы қаншалықты екенін білу үшін, төмендегі сұрақтарға жауап іздейік.

Неге түлектер бірлестігі құрылады? Бұл кімге тиімді?

Бұл әрдайым жағдайларға байланысты: жалпы шешім болмайды. Егер жүйе құрылса, онда нақты тапсырыс беруші – ректор, жоғарғы бітірушілер немесе тіпті студенттер. Бұл осы идеяға келіп, бірінші кезеңде барлық жұмысты нақты мүдделерге қарай шағын адамдар тобы жасайды. Мұндай заттардың мақсаты әрқашан бірнеше: біріншісі – ұрпақтар арасында байланыс орнату, екіншісі – түрлі факультеттер, институттар, қала берді, мамандықтардың түлектері арасында көлденең байланыс орнату, соның ішінде олардың жеке бизнесін құру. Жалпы түлектер бірлестігі – бұл түлектер арасындағы байланыс.

ЖОО түлектер бірлестігін құруды өз қолына алуы керек пе?

ЖОО-ның негізгі міндеті – студенттерді оқыту, сондықтан бұл мәселеде түлектердің бастамасы, олардың Университетпен байланысқа шығу дайындығы өте маңызды. Бұл екі жақты қозғалыс, оны реттеу үшін бірлестіктер қажет – олар ортақ мүдделерді табуға көмектеседі.

Түлектер бірлестігін құрудың негізгі мәселесі қандай?

Ұйымның тұрақтылығы. Шетелде түлектердің бірлестігі не үшін қажет әрі оны және Университетті неге қаржы жағынан қолдау маңызды деген мәселе туындамайды. Онда қазіргі заманғы әлеуметтік сала мәдениетке және оған қосқан үлесіне қатты байланысты. Оксфорд немесе Гарвард дәстүрлерді, тарихты, команда міндеттерін түсінуін қараңыз. Бізде ұрпақтар байланысының, өзара көмек пен қарым-қатынастың құндылығын қайта жаңғырту – бітірушіге осындай қауымдастықтың пайдасын түсіндіру қажет. Маңызды мәселе – бұл қауымдастықтар мен олардың жобаларын қаржыландыру, өйткені олар бюджеттен тыс көздер есебінен іске асырылған. Олардың бірі Университеттің немесе қауымдастықтың мақсатты капиталының қоры болуы мүмкін. Бұл ретте, қорды толықтырумен айналысу қажет және түлектердің бір жолғы қайырмалдықтары түсімдердің жалғыз арнасы болмауы тиіс. Көп нәрсе қор және қауымдастық басшылығының фандрайзингке қызығушылығы мен тартылуына байланысты.

Түлектер бірлестігінің мақсаты – түлектерді серпінді және мүдделі қоғамдастыққа біріктіру:

– түлектердің Университетпен және өзара қарым-қатынастарының үздіксіз байланысы;

– түлектер мен студенттерді оқу орнының қазіргі жағдайы, даму болашағы, оның ғылыми қызметі және басқа да жетістіктері туралы хабардар ету;

– Университеттің одан әрі дамуының негізгі бағыттарын қалыптастыру және осы дамуға жәрдемдесу, Университеттің басқару жүйесі

мен құрылымын жетілдіру, оның беделін нығайту және аймақтың, елдің, жалпы қоғамның аса маңызды мәселелерін шешуге ықпал ету мәселелерінде қауымдастық пен оның мүшелерінің ықтимал қатысуының нысандары мен әдістерін анықтау;

– студенттік ортада жоғары оқу орны үшін мақтаныш сезімін, еліміздің және әлемнің ең үздік жоғары оқу орындарының біріне, Қазақстан қоғамының элитасына қатыстылығын тәрбиелеу, оның түлектерінің үздік жетістіктерін насихаттау;

– қауымдастық ішінде және Университетпен кәсіби, шығармашылық және ғылыми байланыстарды дамыту;

– Университеттің бәсекеге қабілеттілігін арттыру;

– бітірушілердің білім беру процесін ұйымдастырудың тиімді нысандарын дамытуға қатысуы;

– түлектер туралы ақпарат және сол ақпаратқа түлектердің өздерінің қашықтықтан қол жеткізуі;

– Университет түлектерінің бірлескен ынтымақтастығын нығайту.

## **1.2 Жобалаудың мақсаты мен міндеттерін анықтау**

Дипломдық жобаның мақсаты – Университеттің түлектеріне арналған бірлестіктің веб-бағытталған жүйесін, яғни бірыңғай ақпараттық кеңістік және жалпыға ортақ әлеуметтік желіні құру болып табылады.

Дипломдық жобаны құру алдында тұрған негізгі міндеттер келесідей:

– түлектердің Университетпен және өзара қарым-қатынастарының үздіксіз байланысы;

– бірлестік ішіндегі және Университетпен кәсіби, шығармашылық және ғылыми байланыстарды дамыту;

– түлектер туралы ақпарат және сол ақпаратқа түлектердің қашықтықтан қол жеткізуі;

– студенттік ортада жоғары оқу орны үшін мақтаныш сезімін ояту, еліміздің ең үздік жоғары оқу орындарының біріне, Қазақстан қоғамының элитасына қатыстылығын тәрбиелеу, қала берді оның түлектерінің үздік жетістіктерін насихаттау.

Оларды шешудің ең тиімді тәсілі – осы мүмкіндіктерге ие болу үшін және сол қызметтерді көрсету бойынша веб-бағытталған жүйе құру. Және осы жерден веб-бағытталған жүйенің міндеттері айқындалды.

Жүйе арқылы қол жетімді:

– түлектер туралы, соның ішінде, байланыс деректері, мекен-жайы, қазіргі жағдайы және жұмыс орны туралы ақпарат;

– анықтамалық ақпарат, яғни Университетте оқығаны жайлы анықтама, диплом және т.б.;

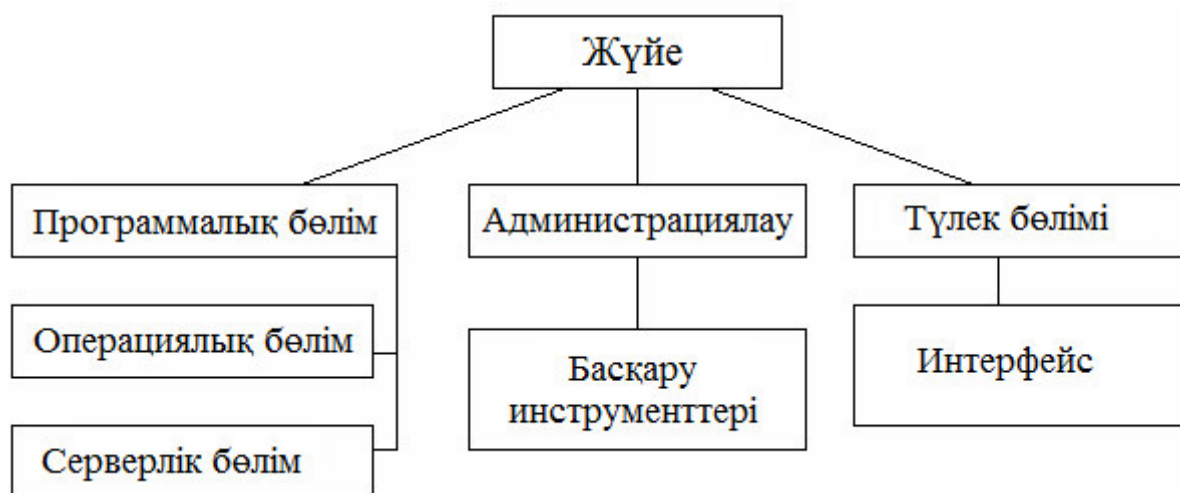
– хабарлама жіберу арқылы қашықтықтан тіркелу, жүйеге кіру;

– Университет жайлы соңғы әрі өзекті ақпарат;

- түлектер кездесуі, маңызды іс-шараларға шақырту алу;
- Университетке инвесторлық көмек.

Қазақстан Республикасының Конституциясының 1-бөлімінің 7-бабында ҚР-дағы мемлекеттік тіл - қазақ тілі, мемлекеттік ұйымдарда және жергілікті өзін-өзі басқару органдарында орыс тілі ресми түрде қазақ тілімен тең қолданылады және мемлекет Қазақстан халқының тілдерін үйрену мен дамыту үшін жағдай туғызуға қамқорлық жасайды деп көрсетілген [1], сондай-ақ ҚР-ның білім беру жүйесі үштілділікке негізделетіндігін ескере отырып, жүйені қазақ, орыс және ағылшын тілдерін қолдайтындай етіп құру басты міндеттердің бірі болды.

Жүйе ақпараттық, артық немесе қажетсіз ақпаратты қамтымауы тиіс әрі архитектурасы қарапайым және интуитивті ыңғайлы болуы керек және 1.1-суретте көрсетілгендей клиенттік бөліктен, программалық бөліктен және администрациялаудан тұрады.



1.1-сурет – Жүйе архитектурасы

Жүйе архитектурасының программалық бөлімі операциялық және серверлік бөлімнің өзара байланысы ретінде қарастырылады.

Операциялық бөлімде жүйені әзірлеу қарастырылған. Болжам бойынша, жүйе PHP скрипттарынан тұрады.

Серверлік бөлім жүйені құру кезінде қолданылатын технологияларды қолдайтын провайдер сайтында жүйені орналастыруды қамтиды.

### 1.3 Жобалық шешімдерді талдау және негіздеу

Қазіргі таңда нарықта жүздеген программалық өнімдер ұсынылып, күн сайын жаңа өнімдер пайда болады. Веб-әзірлеу кезінде жауапкершіліктің бірнеше бағыттарынан таңдау керек: front-end, back-end, full-stack. Front-end әзірлеушілер сайтты жүктеу жылдамдығына және кодтың дұрыс жұмыс жасалуына жауапты, back-end - сервердің кодын жазу үшін, ал full-stack

мамандар тұтынушының барлық талаптарына толық жауап бере алады. Бүгінгі күні full-stack әзірлеушілер еңбек нарығында сұранысқа ие.

Front-end әзірлеудің үш алыбы JavaScript, HTML және CSS болып табылады. Сонымен қатар, соңғы интернет-үрдістерді қадағалап және оларды күнделікті жұмыста қолдана білу қажет. PHP – back-end мамандарға арналған негізгі базалық білімдер жиынтығы. PHP бұл саладағы жалғыз құрал болмаса да, барлық back-end дамудың негізі болып табылады [7]. Екінші тіл ретінде Ruby немесе Python үйренуге болады. Деректер қоры, JavaScript және SQL-мен жұмыс істеу тәжірибесі де қажет. Программалау тілдеріне қосымша, қоса тіркелген барлық қондырмаларды білу маңызды.

Программалау тілін таңдау келесі факторларға негізделуі тиіс:

- еңбек нарығындағы қажеттілік;
- зерттеудің жеңілдігі;
- ұзақ мерзімді перспективалар;
- бұл тілде тағы қандай жобаларды жасауға болады (тіл мен программалау ортасын таңдау).

Түлектер бірлестігінің веб-сайтын әзірлеу алдында шешімнің екі нұсқаны қарастырайық:

– дайын программалық аппараттық құралды сатып алу, мысалы, CMS Wordpress, CMS Joomla сияқты;

– PHP программалау дағдыларын пайдалана отырып сайтты әзірлеу.

Бірінші нұсқада мынадай оң жақтары бар:

– проблемалық бағыттағы жұмыстардағы авторлардың үлкен тәжірибесі;

– салыстырмалы түрде аз уақытта іске асырылады;

– бұл салада тәжірибесі аз адамдар үшін өте маңызды емес мәселелерді шешу.

Дайын программалық құралды пайдаланудың теріс жақтарын сипаттаймыз:

– белгілі бір проблемалық жағдайға үйренудің күрделілігі;

– әмбебап шешімдер, яғни біз функционалдық қызығушылықтар үшін көбірек төлейміз.

PHP программалау дағдыларын пайдаланудың оң жақтары:

– ең аз ақшалай шығындар;

– шынайы объектіге біршама жақындау;

– қызмет көрсету және программаны жаңарту.

Теріс жақтары:

– орындалу мерзімі ұзақ;

– әзірлеуші басқа жұмысқа ауысқан жағдайда программаның қызмет көрсетуі және сервис күрделілігі.

Екі нұсқаны талдап, PHP программалаудағы өзімнің білімімді, дағдыларымды пайдалана отырып, программаны әзірлеу туралы шешім қабылдадым.

Таңдау негіздемесіне жауап беру үшін PHP арқылы сайт программалаудың басқа тілдерімен салыстырып көрелік. Оның негізгі бәсекелестері – Perl, ASP.NET және Java.

**PHP және Perl.** Perl тілін программист Ларри Уолл (Larry Wall) жасап, атауын Practical Extraction and Report Language деп шешті. Мұны «деректерді іс жүзінде алу және есептерді жасау тілі» деп аударуға болады. Perl интерпретатор болып табылады және Web-ке қарағанда әлдеқайда ерте пайда болды. Интернеттің пайда болуымен Perl тілі динамикалық Web-беттерді жасау үшін ыңғайлы құрал болды. Perl әрбір Web-серверде қол жетімді болды, әрі барлығы UNIX басқаруымен жүзеге асырылатын, және баламасы болып тек одан да күрделі C тілі есептелінетін [2].

PHP де Perl-ге ұқсас функционалдылықты ұсынады, бірақ синтаксис, қарапайым. Perl тілі әр түрлі қосымшалар үшін құрылғандықтан, бұл оның синтаксисіне әсер етпеуі мүмкін емес, ал PHP бастапқыда интернет үшін әзірленген. PHP немесе Perl-ді бастапқыда білмеген көптеген пайдаланушылардың пікірі бойынша, PHP тілін үйрену әлдеқайда оңай.

**PHP және Java.** Жалпы айтқанда, Java тілі мен Java технологиясы бірдей емес. Java тілі – «жақсартылған C++» ретінде әзірленген C тіліне ұқсас тіл. Java технологиясы клиенттік және серверлік бөлікті, сондай-ақ деректер қорына қатынауды қамтиды, сондықтан Java технологиясын Apache/PHP/MySQL-мен салыстыру неғұрлым дұрыс. Java технологиясы корпоративтік ауқымдағы Web-сайттарды құруға мүмкіндік беретін платформааралық технология ретінде әзірленген [3]. Бұл технологияның негізгі артықшылығы – платформа аралық төзімділік және объектілі-бағытталған тіл болып табылады, бұл күрделі және көлемді қосымшаларды жасауға мүмкіндік береді. Кемшіліктердің қатарына баяу орындау, жадының көп мөлшерін алу (платформааралық төзімділік үшін төлем) және PHP-мен салыстырғанда Web-қосымшаларды әзірлеу күрделілігін жатқызуға болады. Сонымен қатар, PHP-да жасалған қосымшалардың икемділігі мен ауқымдығы Java-дан кем емес. Сонымен қатар, Java-да қарапайым қосымшаларды әзірлеудің ақталуы екіталай, себебі күрделі міндеттерімен жұмыс істеу кезінде Java қымбатырақ болады.

**PHP және ASP.NET.** ASP (Active Server Pages) - Microsoft сценарий тілі. Егер ASP тілі көптеген параметрлер бойынша, бірінші кезекте, сценарийлерді орындау уақыты бойынша PHP-дан едәуір төмен болса, онда ASP.NET технологиясы пайда болғанда жағдай өзгерді.

ASP.NET-тың басты артықшылығы – бұл веб-сайтты әзірлеу үшін Windows интерфейсінің барлық қуатын пайдалану мүмкіндігі [4]. Бұл динамикалық беттерде пайдаланылатын айнымалы мәндерді сақтау, шифрлау және жіберу туралы алаңдамастан, күрделі интерфейсін бар беттерді жасауға мүмкіндік береді.

**PHP және ColdFusion.** ColdFusion пакеті Allaire фирмасымен әзірленді және дерек қордан алынған ақпаратты өңдеу арқылы интерактивті және динамикалық Web-құжаттарды жылдам дамытуға арналған [5]. ColdFusion-

ның кемшілігі – салыстырмалы түрде төзімділігі төмен. PHP барлық платформаларда жұмыс істейді, ал ColdFusion - тек төртеуінде: Win32, Solaris, HP/UX және Linux. Сонымен қатар, ColdFusion, ASP сияқты коммерциялық болып табылады және өте аз таралған. ColdFusion артықшылығы – әзірлеу ортасы жақсы интеграцияланған, тіл конструкциялары қарапайым және кодты жылдам жазу мүмкіндігі. Рас, тілдің осындай жоғары деңгейіне байланысты ондағы кейбір міндеттер қиынға соғады.

#### 1.4 Веб-бағытталған жүйе құрылымы

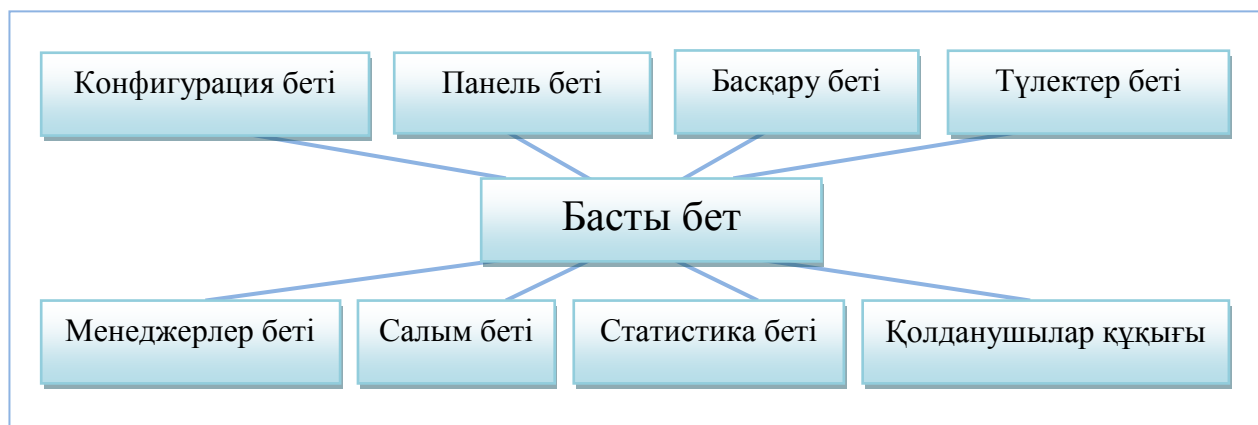
Сайттың дұрыс құрылымы – бұл нақты қалыптастырылған логикалық схема бойынша сайт беттерінің орналасу жүйесі. Сайттың ұрылымын барлық беттердің иерархиясы, олардың қандай да бір каталогтар мен папкаларға тиістілігі сияқты белгілеуге болады.

Сайт құрылымын былай сипаттауға болады:

- мен қазір қайдамын (кіру беті);
- қайда бара аламын (өту);
- категориялар арасындағы өзара байланыс.

Құрылымды қалыптастыра отырып, әлеуетті келушілердің ниеттерін, олар не іздейді және оларды қандай ақпарат қызықтыратынын түсіну керек

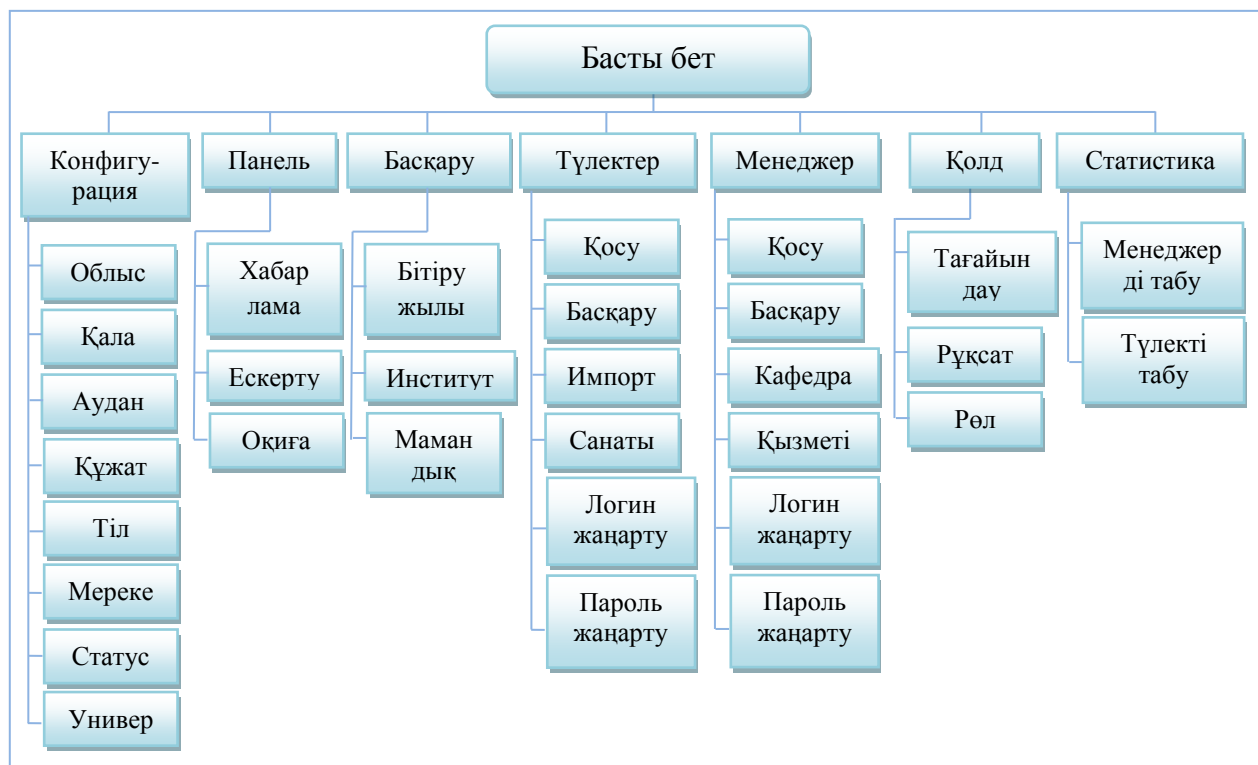
Төменде Университет түлектерінің бірлестігіне арналған веб-бағытталған жүйенің админ бөлімінің құрылымының көрнекі үлгісі көрсетілген (1.2-сурет). Әрбір беттің тақырыбы өте жақын шағын сайттар үшін қолайлы. Мысалы, басты бетте автор туралы ақпарат және қызметке тапсырыс беру нысаны бар фотограф портфолиосы. Ішкі беттерде портфолио, түрлі суреттер орналасады.



1.2-сурет – Жүйенің админ бөлімінің құрылымы

Егер басты беттен басқа 20-30 немесе 100 бет құруды жоспарласаңыз, міндетті түрде сайттың құрылымын жасаңыз, бетті және болашақ беттерді группаларға бөліңіз, ол сайттың әрі қарай өрлеуіне оң әсерін тигізеді.

Сайттың ағаш тәрізді құрылымы ең кең таралған және дұрыс болып табылады, ол көптеген ірі сайттарда, интернет дүкендерде және порталдарда қолданылады. Жүйенің админ бөлімінің сыртқы түрі 1.3-суретте көрсетілген.



1.3-сурет – Жүйенің админ бөлімінің ағаш тәрізді құрылымы

Сайттың сыртқы құрылымы туралы өте ұзақ айтуға болады – пайдаланушылардың ыңғайлылығы, дизайнның бояуы және т.б. Бірақ олардың барлығы бір ережеге бағынады, себебі интернеттің барлық пайдаланушылары барлық сайттардағы блоктардың орналасуының стандартты формасына үйреніп алған (1.4-сурет).



1.4-сурет – Сайттың сыртқы құрылымы



Сайтқа элементтердің базалық орналасуын қарастырайық:

Бас (header) – сайттың жоғарғы бөлігінің атауы, эмблема, негізгі көлденең мәзір және осы сайтқа немесе оның тақырыбына тән қосымша элементтер орналасқан блок (банерлер, жарнама, түсті палитралар, үлкен сурет).

Сол және оң сайдбар (l.sidebar, r.sidebar), тек біреуі немесе екеуі де болуы мүмкін, бұл мәзірдің болмауы тәжірибеде сирек кездеседі. Бұл блоктардың басты мақсаты қосымша ақпарат беру, келушіні сайттың тереңіне жіберу, слогандар мен бейнелерді қызықтыру. Қысқаша сайтқа келушіні мүмкіндігінше ұзағырақ ұстау.

Орталық блок (content), ең басты, бұл ақпарат немесе сервис, ол үшін келуші сіздің сайтыңызға келді. Бұл, ең алдымен, пайдаланушыны қызықтырады. Мазмұн бірінші экранда оңай оқылып, байқалуы керек, ал екіншісінде максимум. Келушіні төменде және төменге қазір не қажет екенін іздеудің қажеті жоқ. Қазіргі уақытта адамдар күтуді ұнатпайды, сіздің сайтыңызды оңай жаба салады да басқасына барады.

Footer, сайттың жертөлесі, блок бірінші кезекте байланыс ақпаратын (компанияның атауы, телефон, мекен-жайы) орналастыруға арналған. Сондай-ақ, жертөле навигациялық мәзір және басқа пайдалы сілтемелерді қосу үшін белсенді пайдаланады.

Сайт беттері – HTML тілінде жазылған мәтіні бар файлдар. Бұл файлдар, оның компьютеріне кіруші бола отырып, браузермен өңделеді және оның бейнелеу құралына шығарылады (монитор, принтер немесе сөйлеу синтезаторы). HTML тілі мәтінді пішімдеуге, функционалдық элементтерді ажыратуға, гипермәтіндік сілтемелерді (гиперсілтемелер) жасауға және көрсетілген бетке бейнелерді, дыбыс жазбаларды және басқа да мультимедиялық элементтерді қоюға мүмкіндік береді. Бетті көрсету оған CSS немесе JavaScript тілінде сценарий кестелерін қосу арқылы өзгертуге болады.

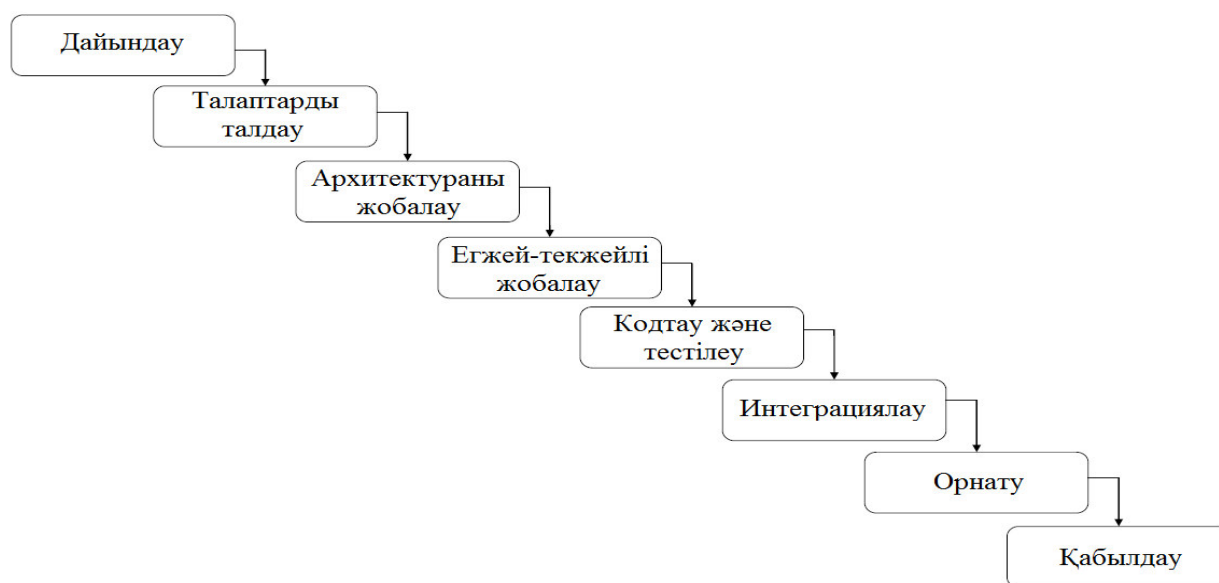
Сайт беттері файлдардың қарапайым статикалық жиынтығы немесе сайт қозғаушы деп аталатын серверде арнайы компьютерлік программамен жасалуы мүмкін. Қозғалтқыш жеке сайт үшін тапсырыс жасалуы мүмкін немесе сайттардың кейбір класына есептелген дайын өнім болуы мүмкін. Кейбір қозғалтқыштар сайт иесіне құрылымдауды икемді күйге келтіру және ақпаратты веб-сайтта шығару мүмкіндігін қамтамасыз ете алады. Мұндай қозғалтқыштар мазмұнды басқару жүйелері деп аталады.

Программалық қамтамасыз етуді әзірлеу процесі (ағылш. software development process – software process) – программалық қамтамасыз етуді әзірлеу құрылымы. Мұндай процестің бірнеше үлгілері бар, олардың әрқайсысы процесс барысында орын алатын міндеттер немесе қызмет түрінде өз тәсілін сипаттайды.

Intellect ақпараттық порталының ұсынған мәліметтері бойынша программалық қамтамасыз етуді әзірлеу үшін үш модель пайдаланылады [27].

**Сарқырама (каскадтық) моделі.** Оның негізгі сипаттамасы – бүкіл дамудың кезеңдерге бөлінуі және бір кезеңнен екінші кезеңге көшу қазіргі жұмыс аяқталғаннан кейін ғана жүреді. Әрбір саты дамудың басқа даму тобымен жалғастыруға болатын жеткілікті құжаттаманың толық жиынтығымен аяқталады (1.5-сурет).

Сарқыраманы пайдалану процесі технологиялық дамудың басында, әзірлеушілерге оларды техникалық тұрғыдан мүмкіндігінше оңтайландыруға мүмкіндік беру үшін барлық талаптар дәл және толығымен тұжырымдалған жүйелер үшін тиімді. Күрделі есеп айырысу жүйелері, нақты уақыттық жүйелер және басқа да осындай міндеттер осы санатқа жатады.



1.5-сурет – Сарқырама моделі

Сарқырама моделіне сәйкес жобаның кезеңдері:

- талаптарды қалыптастыру;
- жобалау;
- іске асыру;
- тестілеу;
- енгізу;
- пайдалану және техникалық қызмет көрсету.

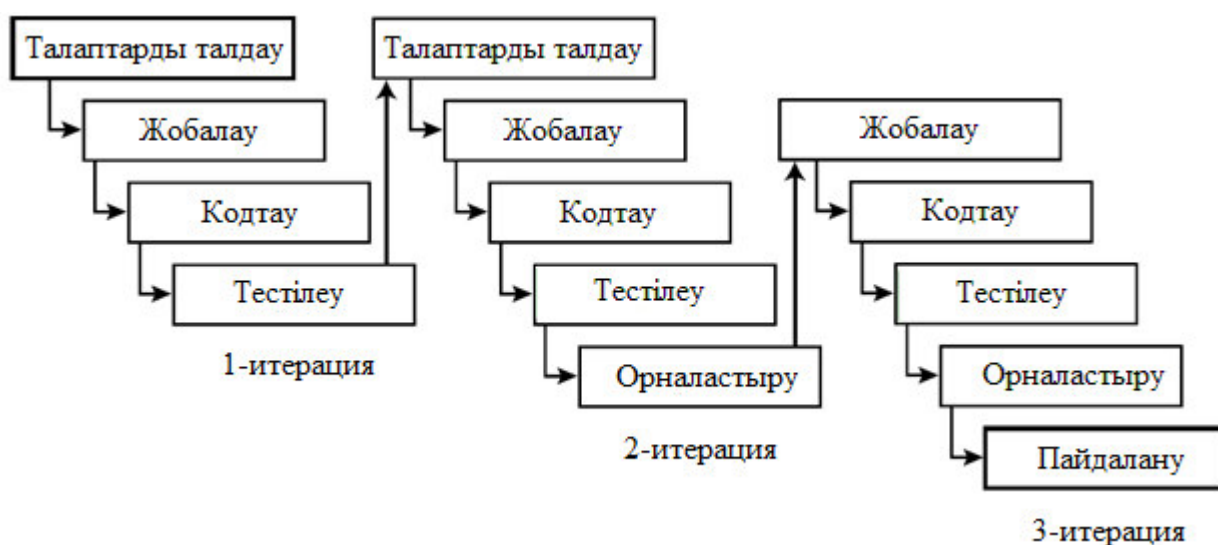
Артықшылықтары:

- әр кезеңде толық және дәйекті құжаттама;
- жобаның уақытын және құнын анықтау оңай.

Кемшіліктері. Сарқыраманың моделінде, бір жобаның фазасынан екіншісіне көшу алдыңғы фазаның нәтижесін (шығуын) толық дәлелдейді. Дегенмен, талаптың дұрыс еместігі немесе оның дұрыс түсіндірілмеуі жобаның бастапқы кезеңіне «кері оралу» керек және қажетті қайта өңдеу жобаны кестеден бұзып қана қоймай, көбіне шығындардың сапалы өсуіне әкеледі жобаның аяқталуына, ол бастапқыда ойластырылған түрінде

шығарылды. Заманауи мамандардың пікірінше, сарқырама моделінің авторларының негізгі қателігі - бұл жоба бір рет бүкіл процесс арқылы жүретінін болжау, жобаланған архитектура жақсы және оңай, іске асыру жобасы ақылға сыйымды және тестілеу кезінде іске қателер оңай жойылады. Бұл модель барлық қателерді іске асыруда шоғырландырылатын болады деп болжайды, сондықтан оларды жою компоненттер мен жүйені тестілеу кезінде біркелкі жүреді. Осылайша, ірі жобалар үшін сарқырама моделі өте шынайы емес тек шағын жүйелерді құру үшін тиімді қолданылуы мүмкін.

**Итерациялық модель.** 1970 жылдарда Iterative and incremental development model (ағылшын итеративті және инкрементальдық даму, IID) деп Т. Гилба атаған эволюциялық модельдің атауы болып табылады. Сондай-ақ, бұл модель итерациялық модель және қосымша модель деп аталады (1.6-сурет).



1.6-сурет – Итеративті модель

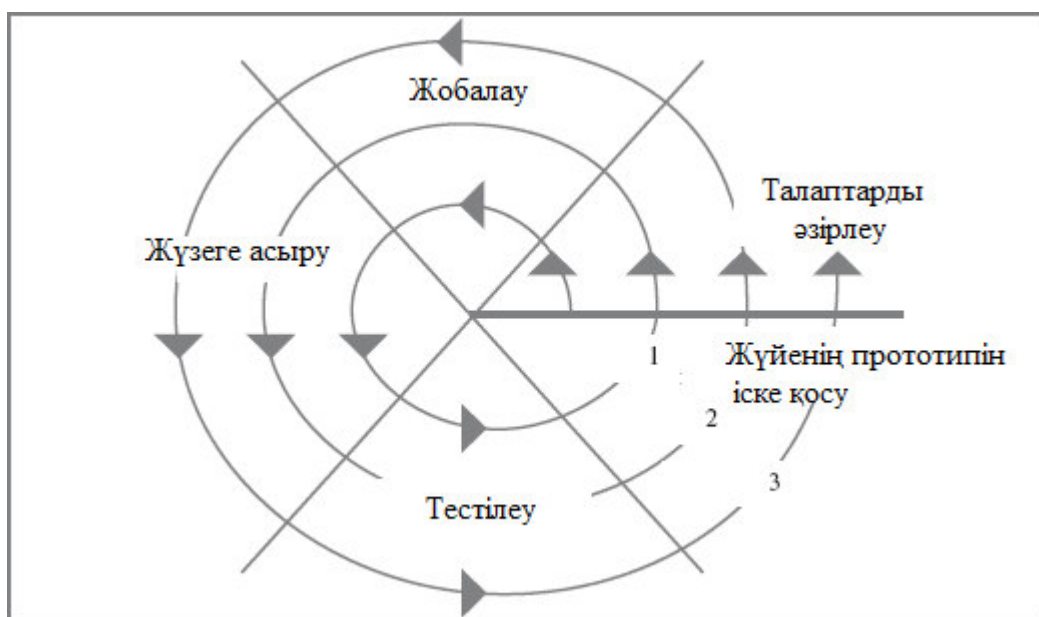
IID моделі жобаның өмірлік циклін жобаның өмірлік циклін жобаның өмірлік циклін бөлуді қамтиды, олардың әрқайсысы тұтастай алғанда жобаға қарағанда функционалдылықтың аз бөліктерін жасау үшін қолданылатын барлық даму процестерін қоса алғанда, олардың әрқайсысы «шағын жобаға» ұқсас. Әрбір итерацияның мақсаты – барлық алдыңғы және ағымдағы итерациялардың интеграцияланған мазмұны арқылы анықталған функционалдылықты қоса, программалық жасақтаманың жұмыс нұсқасын алу. Соңғы итерацияның нәтижесі өнімнің барлық қажетті функционалдығын қамтиды. Осылайша, әрбір иерархия аяқталғаннан кейін, өнім эволюциялық дамып келе жатқан қабілеттеріне қарай өседі. Бұл жағдайда итерация, инкрементті және эволюциялық әртүрлі сөздердегі бірдей мағынаны әр түрлі көзқарастарды білдіру болып табылады.

Т.Гилбаның айтуынша, «эволюция - тұрақтылықтың пайда болуын қамтамасыз ететін құрылғы. Кешенді жүйені сәтті құру мүмкіндіктері шағын

қадамдар сериясымен жүзеге асырылса, және әрбір қадам айқын анықталған табысқа ие болса, сондай-ақ сәтсіздікке байланысты бұрынғы табысты қадамға «оралу» мүмкіндігі бар болса, барынша күшейтілетін болады. Жүйені құруға арналған барлық ресурстарды пайдалануға енгізгенге дейін, әзірлеуші нақты әлемнен кері байланыс сигналдарын алуға және жобадағы қателіктерді түзеуге мүмкіндік алады.

ИД-тің көзқарасы оның жағымсыз жағына ие. Біріншіден, жобаның мүмкіндіктері мен шектеулерін біртұтас түсіну ұзақ уақыт бойы болмады. Екіншіден, итерация кезінде бұрын атқарылған жұмыстардың бір бөлігін қабылдамау керек. Үшіншіден, жұмысты орындаудағы мамандардың адалдығы әлі де төмендейді, бұл психологиялық тұрғыдан түсінікті, өйткені олар үнемі «бәрі қайтадан жаңарып, кейінірек жетіле алады» деген сезіммен жүреді. Көптеген заманауи даму әдістемелерінде (RUP, MSF, XP) итерациялық тәсілдің әртүрлі нұсқалары енгізілген.

**Спираль моделі.** Модель 1980-ші жылдардың ортасында Бэрри Бехаммен әзірленген. Ол классикалық Деминг PDCA (plan-do-check-act) циклына негізделген. Бұл модельді пайдаланған кезде программалық жасақтама бірнеше итерацияда (спираль бұрылыстарында) прототиптеу әдісімен жасалады (1.7-сурет).



1.7-сурет – Спираль моделі

Әрбір итерация фрагменттің немесе программалық қамтамасыз етудің нұсқасына сәйкес келеді, ол жобаның мақсаттары мен сипаттамаларын анықтайды, нәтижелердің сапасын бағалайды және келесі итерацияның жұмысын жоспарлайды. Әрбір итерация кезінде төмендегілер бағаланады:

- жобаның шарттары мен құнынан асып кету қаупі;
- басқа итерацияны орындау қажеттілігі;

- түсіну жүйесінің талаптары толықтығы мен дәлдігі;
- жобаның аяқталуының орындылығы.

Спираль моделі эволюциялық модельге (IID үлгісі) балама емес арнайы әзірленген нұсқа екенін түсіну маңызды. Өкінішке орай, спираль моделі жиі эволюциялық модель синонимі ретінде қабылдайды.

Спираль үлгісінің айрықша ерекшелігі өмірлік циклді және бақылау нүктелерін ұйымдастыруға әсер ететін тәуекелдерге ерекше назар аударады. Боэм басымдығы бойынша ең көп таралған 10 тәуекелді тұжырымдайды:

- мамандардың жетіспеушілігі;
- шындыққа жанаспайтын мерзім мен бюджет;
- қолайсыз функционалдылықты жүзеге асыру;
- дұрыс емес, қолайсыз интерфейсті әзірлеу;
- бөлшектерді қажетсіз оңтайландыру және нақтылау;
- үздіксіз өзгерістер;
- жүйенің ортасын анықтайтын немесе интеграцияға қатысатын сыртқы компоненттер туралы ақпараттың жетіспеушілігі;
- сыртқы (жобаға қатысты) ресурстармен жұмыс жасаған кездегі кемшіліктер;
- алынған жүйенің жеткіліксіз өнімділігі;
- әртүрлі саладағы мамандардың біліктілігіндегі айырмашылық.

Қазіргі спираль моделінде келесі бақылау нүктелерінің жиынтығы анықталды:

- Concept of Operations (COO) – жүйенің тұжырымдамасы;
- Life Cycle Objectives (LCO) – өмірлік циклдің мақсаты мен мазмұны;
- Life Cycle Architecture (LCA) – өмір циклінің архитектурасы, мұнда программалық жасақтаманың тұжырымдық архитектурасының дайындығы туралы айтуға болады;
- Initial Operational Capability (IOC) – бастапқы операциялық мүмкіндіктер;
- Final Operational Capability (FOC) – нақты жұмыс істеуге арналған (дайындалған және конфигурацияланған) дайын өнім.

Дж. Гарреттің пікірінше, шынымен пайдалы сайт тұтынушының талаптарын қанағаттандыра отырып, компанияның стратегиялық мақсаттарына қызмет етуі тиіс [28]. Гарреттің "Тұтынушылық іс-әрекеттер дизайны" моделі 1.8-суретте көрсетілген. Сайтты әзірлеушілердің бір бөлігі веб-сайтты программалық өнім ретінде қарастырады және жұмыс станциялары мен серверлері үшін дәстүрлі әзірлеушілер арасында прогараммалық қамтуды қолданады. Ал екінші бөлігі сайттың интернетте кеңірек таралуын және ақпарат алуын қамтамасыз етеді.

Бұл моделді жобалау бес кезеңнен тұрады.

- стратегиялық – веб-сайттың мақсаттары мен тұтынушылардың талаптары анықталады;

- мүмкіндіктерді жинау – сайттың мазмұнына қойылатын талаптар, функционалдық сипаттамалар және біліктілік талаптары қарастырылады;
- құрылымдық – сайттың құрылымы мен ақпараттық архитектураны анықтау;
- пішімдеу – интерфейсті жобалау, навигация, ақпараттық дизайн және беттің прототипі орындалады.
- жоғарғы кезең – визуалды безендіруді, контрасты, ішкі және сыртқы келісуді, түс балансын және қаріпті таңдау.

Ғаламтор бағдарламалық интерфейс ретінде		Ғаламтор гипермәтінді ақпараттық кеңістік ретінде	
Жоғарғы бөлім	Визуальді дизайн		
Форматтау кезеңі	Интерфейс дизайні	Навигация дизайні	
	Ақпараттық дизайн		
Құрылымдық кезең	Іс-әрекетті жобалау	Ақпараттық архитектура	
Мүмкіндіктер жиыны	Функционалдық сипаттама	Контентке деген талап	
Ғаламтор бағдарламалық интерфейс ретінде		Ғаламтор гипермәтінді ақпараттық кеңістік ретінде	
Стратегиялық кезең	Қолданушылар талабы		
	Сайт мақсаты		

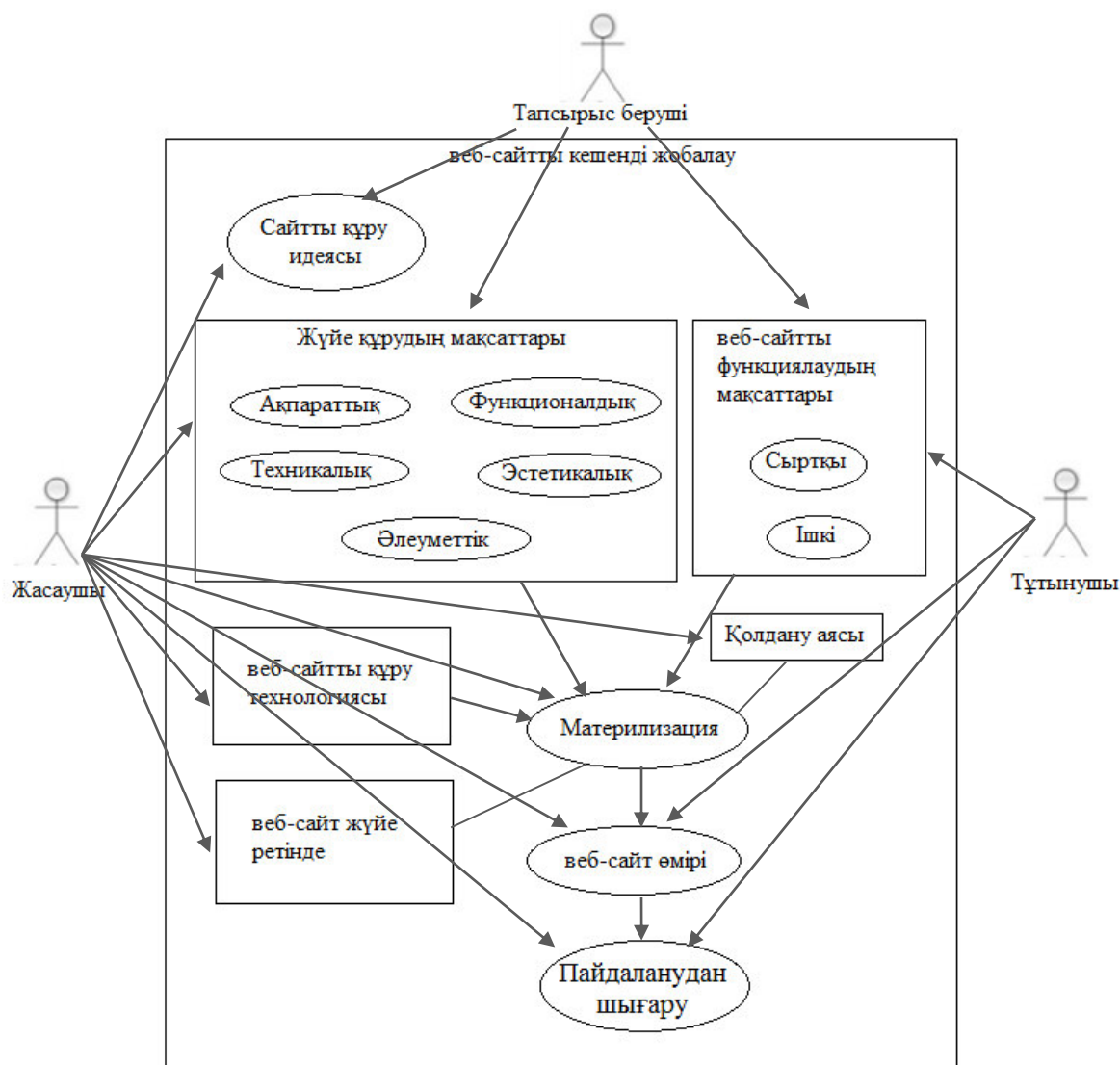
1.8-сурет – Дж. Гарреттің "Тұтынушылық іс-әрекеттер дизайны" моделі

**Программалық жасақтама әзірлеу әдістемесі.** Әдістеме – бұл принциптердің жүйесі, сондай-ақ программалық қамтамасыз етудің даму стилін анықтайтын идеялар, тұжырымдамалар, әдістер және құралдар жиынтығы [10].

Әдістеме – стандарттарды енгізу. Стандарттар тек қана не болу керектігін көрсетеді, ал таңдау еркіндігі программистке қалдырылған.

Таңдалған әдіс арқылы нақты заттар жүзеге асырылады. Бұл дамудың қалай жүргізілетінін анықтайды. Программалық қамтамасыз етуді жетілдірудің көптеген әдістемелері бар. Нақты әдістемені таңдау өлшемдерге, жобаның ерекшелігі мен күрделілігіне, компаниядағы процестердің тұрақтылығы мен жетілуіне және қызметкерлердің жеке қасиеттеріне

байланысты. Әдістемелер программалық қамтамасыз етуді дамытудың басқару теориясы болып табылады.



1.9-сурет – Веб-сайтты құру процесіне қатысушылар арасындағы байланыс

Мәтіннің оқуға жеңілдігі оқу жылдамдығын және оның қабылдануын аттырады, тұтынушының да мәтінді аяқтып оқу ықтималдығыды артады. Оқу жеңілдігіне қаріп, қаріп көлемі, жол биіктігі, сөздер арқашықтығы және азат жол секілді факторлар әсер етеді. Зерттеу нәтижесі бойынша шегініссіз мәтіндер тез оқылады, алайда қабылдау жағы азаяды. Сайтты тұтынушылар басты беттен көп бастай бермиді. Іздеу сайтта маңызды фактор болып табылады.

Бастапқыда болашақ сайттың иесі оның негізгі мақсаттары мен міндеттерін анықтайды, атауын ойлап табады, домендік атауды таңдайды. Көп жағдайда сайт иесі өз бетінше сайт жасай алмайды, сондықтан ол тиісті мамандарға жүгінеді. Бұл ұйым (веб-студия) және жеке тұлға (фрилансер)

болуы мүмкін. Бұл жағдайда сайт иесі тапсырыс беруші ретінде, ал веб-студия немесе жеке тұлға орындаушы ретінде әрекет етеді. Ұйымдармен әрқашан келісім-шарт жасалады, ал жеке тұлғалармен, негізінен, өзара сенімге негізделген ауызша келісім жүргізіледі. Шарттың өзінде тек жалпы заңды сәттер ғана сипатталады, төлеу мерзімдері, тәсілдері және т.б. айтылады. Күрделі жобалар үшін даулы жағдайлар мен түсініспеушіліктерді болдырмау үшін тапсырыс беруші арнайы құжат — техникалық тапсырма дайындайды, онда оны қызықтыратын барлық сәттерді егжей-тегжейлі сипаттайды.

Веб-дизайнерлер веб-бет үлгілерінің макеттерін жасайды. Дизайнер соңғы тұтынушы сайттың ақпараттары мен қызметтеріне қалай қол жеткізетінін анықтайды. Яғни, веб-интерфейсті әзірлеумен тікелей айналысады. Көптеген жағдайларда веб-беттер графикалық элементтерсіз болмайды. Оларды дайындаумен суретшілер мен фотографтар айналысады. Дайын үлгілер тапсырыс берушіге көрсетіледі. Осы сәтте беттер әлі толтырылмаған болуы мүмкін (бұл дизайнердің міндетіне кірмейді). Макеттер көрнекі болу үшін оларға еркін мазмұн орналастырылады. Дизайнерлердің слэнгінде бұл мазмұн балық деп аталады. Егер тапсырыс беруші шаблондардың сыртқы түріне қанағаттанса, онда олар беттеушіге (верстальшыға) беріледі.

Беттеуші (верстальшы) үлгілердің макеттерін суреттер түрінде алады (әдетте PSD пішімінде). Оның міндеті - интернет үшін жеке кесілген және дайындалған бейнелері бар гипермәтіндік веб-беттер жасау. Беттеуші (верстальшы) сондай-ақ мазмұнын орналастыруға міндетті емес, ол тек пішімді өзгертеді. Веб-беттерді көруге арналған көптеген браузерлер-программалармен үйлесімділігін қамтамасыз ету верстальшы жұмысындағы күрделі сәттердің бірі болып табылады. Веб-шолушылар CSS таңбаларының немесе ережелерінің бірдей элементтерін өзінше түсіндіре алады, соның нәтижесінде кейбір пайдаланушылар мазмұнды дизайнерді ойлағандай емес көре алады және Тапсырыс беруші көре алады. Верстальщик көптеген браузерлер үлгілерді бірдей көрсетеді, олар ары қарай беріледі.

Маңызды ресурстар болып табылатын көптеген сайттар бар. Бұл ресурстарда пайдаланушылардың дербес деректері (мысалы, жеке хат алмасу, мекен-жайлар, телефондар) немесе қаржылық ақпарат (мысалы, банктік сайттар) болуы мүмкін. Мұндай ресурстарды бұзу тікелей ақша шығындарына (мысалы, қаскүнем бөтен шоттан өз шотына ақша аудара алады), сондай-ақ жанама, құпия ақпаратты таратумен байланысты немесе жай ғана қаскүнем сайт мазмұнын бүлдіре алады. Көптеген сайттар үшін кейбір қауіпсіздік деңгейін қамтамасыз ету маңызды. Қажетті қауіпсіздік деңгейі сайттағы ақпаратқа байланысты.

Дипломдық жобаны әзірлеу келесі кезеңдерден тұрады:

- а) қолданыстағы шешімдерді талдау;
- ә) сайтты ақпараттық жобалау;
- б) мазмұн жазу;
- в) код жазу;



г) тестілеу.

Екінші негізгі кезеңге немқұрайлы қарау сайттың алға жылжуы мен қолдауы кезіндегі қиындықтарға, сондай-ақ сайттың кейіннен толық қайта өңдеу қажеттілігіне әкелуі мүмкін.

Сайттарға арналған контент ресурстың өзі құрылғанға дейін мұқият жоспарланады. Порталға арналған мазмұн көптеген параметрлерді ескере отырып таңдалады: өзектілігі, тақырыбы, белгілі бір аудиторияға бағытталуы, қолданылуы. Сайт үшін мәтіндік контент "seo", "сайттың семантикалық ядросы", "интернет-маркетинг" және т. б. сияқты ұғымдар туралы түсінікке ие адам жазу керек.

Тестілеу мақсаты – әзірленген шешімнің техникалық тапсырмаға сәйкестігін тексеру және порталдағы барлық тетіктерді ретке келтіру. Сайт беттерінің қолжетімділігін және оның функционалының жұмыс қабілеттілігін тексеру жүргізіледі. Бұл да маңызды кезең. Сайттың программалық кодындағы қателіктер оны іздеу арқылы индекстеудің қиындауына, веб-беттерді жүктеудің баяулауына және соның салдарынан сайтқа кіру және конверсияның төмен болуына әкеп соқтыруы мүмкін.

## 2 Программалық қамтамасыз етуді жобалау

### 2.1 Программаны әзірлеу инструменттері

«Университет түлектерінің бірлестігі» веб-бағытталған жүйесін әзірлеу үшін, жоғарыда талдаудың нәтижесі бойынша PHP тілі таңдалды.

PHP – программалау тілі [8]. Ол негізгі JavaScript сияқты уақыттың бір бөлігін программист мәндермен жұмыс істеуді жүргізді және әрбір жеке алынған уақытта кодта путин қандай жолмен жүру керектігі туралы шешім қабылдайды.

PHP – сценарийлерді жасау тілі. PHP – бұл Personal home Page сөздерінің аббревиатурасы, бірақ қазіргі таңда бұл атау ескірген болып есептеледі. PHP бірінші нұсқасы 1994 жылы Расмус Лордорфпен жасалды және Web-беттерге келушілерді бақылауға арналған құралдар жиынтығы болды. Уақыт өте келе PHP құралдар жиынтығынан программалаудың толыққанды тіліне айналды, ал оның атауы PHP HyperText Preprocessor (PHP гипермәтінінің препроцессоры) рекурсивті білім ретінде өзгертілді.

PHP – сценарийлердің серверлік тілі. HTML-мәтінге салынған PHP конструкциялары әр бетке кіргенде сервермен орындалады. Оларды өңдеу нәтижесі әдеттегі HTML мәтінімен бірге браузерге жіберіледі. PHP ашық бастапқы коды бар және тегін таратылады.

PHP ерекшеліктері [12]:

- веб-жобалардың серпінділігі. PHP базалық мүмкіндіктерінің бірі – HTML-ге бетті белгілеу және оның модификациясы, бұл толық веб-сайттар бір бет қорында "жинауға" мүмкіндік береді;

- ашықтық. Интерпретатор нұсқасы әрқашан GitHub-та қол жетімді;

- жан-жақтылығы. PHP – бұл кроссплатфорлы программалау тілі, оны түрлі операциялық жүйелерде қолдануға болады;

- пайдалану оңай. PHP синтаксисі программист үшін түсіну және пайдалану үшін өте оңай. Тіл көптеген мүмкіндіктерге ие және үлгі ретінде пайдаланылуы мүмкін;

- техникалық қолдау. PHP ресми веб-бетіне (<http://php.net/>) кіруге болады. Оның ішінде талқылау үшін ауқымды ұсынылатын тақырыптар тізімі және арнайы мысалдармен тіл бойынша толық құжаттама қамтылған;

- дербестендіру. PHP ашықтығы программистерге программалық органы интерпретатор өзегі деңгейінде түрлендіруге, оның функционалдығын түрлендіруге мүмкіндік береді. Сонымен қатар, PHP қателердің ықтималдығын азайтуға мүмкіндік береді;

- қауіпсіздік. Егер PHP скрипт дұрыс әзірленген болса, онда оның программалық коды тек құрушыға белгілі болады және оны сырттан түрлендіру мүмкіндігі болмайды;

- көптеген мәтіндік операторлар. PHP-де мәтінге кіруді іздеуге, мәтіннен фрагментті кесуге және мәтіндік мәндермен көптеген басқа

операциялар жасауға мүмкіндік беретін әр түрлі мәтіндік өрнектердің үлкен саны бар;

– типтің міндетті еместігі. PHP-де тілдің техникалық ерекшелігі программаның басында басқа программалау тілдері ретінде деректер түрлерін міндетті түрде жариялауды талап етпейді.

Деректер қоры – ақпаратты сақтауға, оны қажеттілігіне қарай алуға және жүйелеуге мүмкіндік беретін құрал. Оған құжаттарды қоюға болады, одан құжаттарды жүйелеу үшін папкалар мен жапсырмаларды алуға болады.

Жақсы деректер қоры ақпаратты ұзақ уақыт сақтай алады. Ол программа жұмысын тоқтатып немесе бүкіл веб-сервер қайта іске қосылған кезде де ақпаратты жоғалтпайды. Жақсы деректер қоры тұрақты сақтаушы болуы керек. Программалау ұғымдарында бұл ақпараттың сақталуы деп аталады. Басқаша айтқанда, егер веб-сервер тоқтатылса немесе деректер қоры тоқтатылып, қайта іске қосылса да, деректер қорына орналастырылған ақпарат орнында қалады.

MySQL – бұл деректер қоры. Онда кез келген ақпарат, пайдаланушыларымыздың деректері және басқа да қажетті мәліметтер сақталады. MySQL ақпаратты сақтау қабілетіне байланысты өте танымал. Бұл әлемдегі ең кең таралған ДҚБЖ (деректер қорын басқару жүйесі). Миллиондаған пайдаланушылар оның проблемаларын іздеу мен сипаттау және оның мүмкіндіктерінің шекарасын тексеру арқылы онымен жұмыс істейді, және мындаған әзірлеушілер оның кодын жетілдіруге көмектесті. Қысқаша айтқанда, MySQL – кейінірек іздеп табуға болатын заттарды қоюға болатын қойма. Сонымен қатар, MySQL осы қоймада іздеу үшін нақты жылдам механизм ұсынады. Ол PHP үшін тамаша серіктес. Ол кез келген жүйеде оңай орнатылады және ірі коммерциялық ДҚБЖ сияқты үлкен ресурстарды (мысалы, Oracle және IBM өнімдері) талап етпейді [23].

MySQL негізгі артықшылықтары:

– масштабталу. MySQL Yahoo!, Google, HP, Associated Press-тағы қомақты мөлшерлі ДҚ жұмысын қолдай алады;

– төзімділік. MySQL түрлі платформаларда жұмыс істейді, олардың ішінде Unix, Linux, Windows, OS / 2, Solaris, MacOS;

– байланыстылық. MySQL желілік құрылымы бар. MySQL бір уақытта кез келген интернет нүктесінен бірнеше пайдаланушыға кіруге болады. MySQL-дің көптеген программалық интерфейстері бар (Application Programming Interface-API), олар C, C++, Perl, PHP, Java, Python сияқты тілдерде жазылған қосымшалардан MySQL-мен өзара әрекеттесуге мүмкіндік береді;

– қауіпсіздік. MySQL деректерге қатынауды бақылау жүйесі бар және мәліметтерді жіберу кезінде шифрлауды қамтамасыз етеді;

– жұмыс істеу жылдамдығы;

– пайдалану ыңғайлылығы. MySQL өте ыңғайлы орнатылады және іске асырылады, оңай басқарылады;

– ашық код.

PHP сценарийі дерек қорға қосылу қажет болғанда, MySQL-мен байланысады және әдетте дерекқордың атауын және оған қажетті кестені хабарлайды [6].

MYSQL сол атау арқылы белгілі дерек қорды табады. MYSQL дерек қормен және ондағы қажетті кестемен байланысады, пайдаланушының сұрауын өңдейді және өңдеу нәтижелерін сценарийге қайтарады. Ең маңыздысы - әзірлеудің жылдамдығы. MYSQL «программалау тілінің серверлік деректер қорының» басқа буындарымен салыстырғанда ең жылдам болып табылады. Даму жылдамдығы дегеніміз, ең алдымен, жүйені әзірлеудің төмен құны.

Дерек қор, ең алдымен, кестелер жиынтығы. Кесте объектілердің белгілі бір жиынтығының сипаттамалары (атрибууттары) бар екі өлшемді кесте ретінде қарастырылуы мүмкін. Кесте атау мен оған сілтеме жасайтын идентификаторға ие.

Кестенің бағандары нысандардың бір немесе басқа сипаттамаларына өрістерке сәйкес келеді. Әр өріс деректерді сақтаудың атауы мен түрлерімен сипатталады. Өріс атауы әртүрлі программаларда деректерді өңдеу үшін қолданылатын идентификатор болып табылады.

Дерекқор кестелерін құрастырған кезде, ақпараттың дәйектілігін қамтамасыз ету маңызды. Бұл, әдетте, негізгі өрістерді енгізу арқылы жүзеге асырылад. Бір немесе бірнеше өрістер кілтті болуы мүмкін.

Деректер қорын және өңдеу сұрауларын ДҚБЖ (деректер қорын басқару жүйесі) жасайды. Әртүрлі ДҚБЖ дерекқорларды әртүрлі ұйымдастырады және сақтайды. Мысалы, Paradox және dBase әр кестеге арналған бөлек файлды пайдаланады. Бұл жағдайда дерекқор - кесте файлдары сақталатын каталог болып табылады. Microsoft Access және InterBase-те бірнеше кестелер бір файл ретінде сақталады. Бұл жағдайда, дерек қор - оған кіру жолы бар файлдың атауы. Sybase немесе Microsoft SQL серверлері сияқты клиенттік / серверлік жүйе барлық деректерді бөлек компьютерде сақтайды және SQL деп аталатын арнайы тіл арқылы клиентпен байланысады.

Дерекқорлармен жұмыс істегенде, барлық өзгерістерді кәштеу жиі қолданылады. Бұл деректерге барлық өзгерістер, жаңа жазбаларды енгізу, бар жазбаларды жою, яғни, пайдаланушымен орындалатын барлық деректерді өңдеу ең алдымен дерекқорда жасалмаған, бірақ уақытша виртуалды кестеде жадта сақталады. Тек арнайы командадан кейін, деректер кестесіне енгізілген деректердің дұрыстығын тексергеннен кейін, пайдаланушы бұл деректердің барлығын дерекқорға жаза алады немесе оны қалдырып, өңдеуден бұрын мемлекетке оралуы мүмкін.

Деректер қорына өзгерістер енгізу транзакциялар арқылы жүзеге асырылады. Бұл деректер қорын өзгертетін пәрмендер жиынтығы. Мәміле барысында пайдаланушы деректердегі бірденені өзгерте алады, бірақ бұл тек сыртқы көрініс. Шын мәнінде, барлық өзгерістер еске сақталады. Ал пайдаланушыға нақты деректер қорына барлық өзгертулер енгізіп немесе оны

қабылдамай, транзакция басталғанға дейін мемлекетке оралу арқылы мәмілені аяқтау мүмкіндігі беріледі.

Деректер қорын жасау үшін әзірленген программа шеңберінде MySQL таңдалды.

**SQL негіздері.** SQL қалай жұмыс істейтінін қарастырайық. Мысалы, ДҚБЖ басқаратын деректер қоры бар. Одан деректерді шығару үшін, SQL-де жасалған сұрауды пайдаланады. Бұл сұрауды өндейтін ДҚБЖ сұратылған деректерді шығарып, оны қайтарады. Немесе көрсетілген параметрлермен деректерді қосу, жаңарту, жою үшін әрекеттерді орындайды.

**MYSQL операторлары.** MYSQL деректер қорында әртүрлі әрекеттерді орындау үшін шамамен 40 оператор бар. Төменде осы жұмыста қолданылатын негізгі операторлар бар.

**Деректерді анықтау тілі (DDL).** Деректерді анықтау тілінде дерекқорлар мен олардың ішіндегі нысандарды (кестелер, көріністер және т.б.) жасау, өзгерту және жоюға мүмкіндік беретін операторлар бар. Бұл операторлар 2.1-кестеде келтірілген.

2.1-кесте – MYSQL операторлары

Оператор	Сипаттамасы
CREATE TABLE	Деректер қорына жаңа кесте қосу үшін пайдаланылады.
CREATE DOMAIN	Жаңа домен жасау үшін пайдаланылады.
CREATE BASE	Деректер қорын жасау үшін қолданылады

**Деректерді манипуляциялау тілі (DML).** Деректерді манипуляциялау тілінде деректерді таңдауға, қосуға, жоюға және өзгертуге мүмкіндік беретін операторлар бар. Есіңізде болсын, бұл операторлар талап етілмеген мәмілені аяқтау үшін талап етілмейді. DML операторлары 2.2-кестеде келтірілген.

2.2-кесте – MYSQL операторлары

Оператор	Сипаттамасы
SELECT	Деректерді таңдау үшін пайдаланылады
INSERT	Кестеге жолдарды қосу үшін қолданылады.
DELETE	Кестеден жолдарды жою үшін пайдаланылады.
UPDATE	Деректерді өзгерту үшін пайдаланылады

Кейде SELECT операторы Data Query Language (DQL) деп аталатын бөлек санатқа түседі.

Мәмілені (транзакцияны) басқару тілі (TCL). Мәмілені басқару тілінің операторы DML операторының тобы жасаған өзгерістерді басқару үшін қолданылады. TCL операторлары 2.3-кестеде көрсетілген.

### 2.3-кесте – MYSQL операторлары

Оператор	Сипаттамасы
COMMIT	Мәмілені аяқтау және дерекқорға өзгерістерді сақтау үшін пайдаланылады.
ROLLBACK	Мәмілені қайтару және дерекқорға өзгертулерді болдырмау үшін пайдаланылады.
SET TRANSACTION	Ағымдағы мәмілеге деректерге қатынау параметрлерін орнату үшін қолданылады.

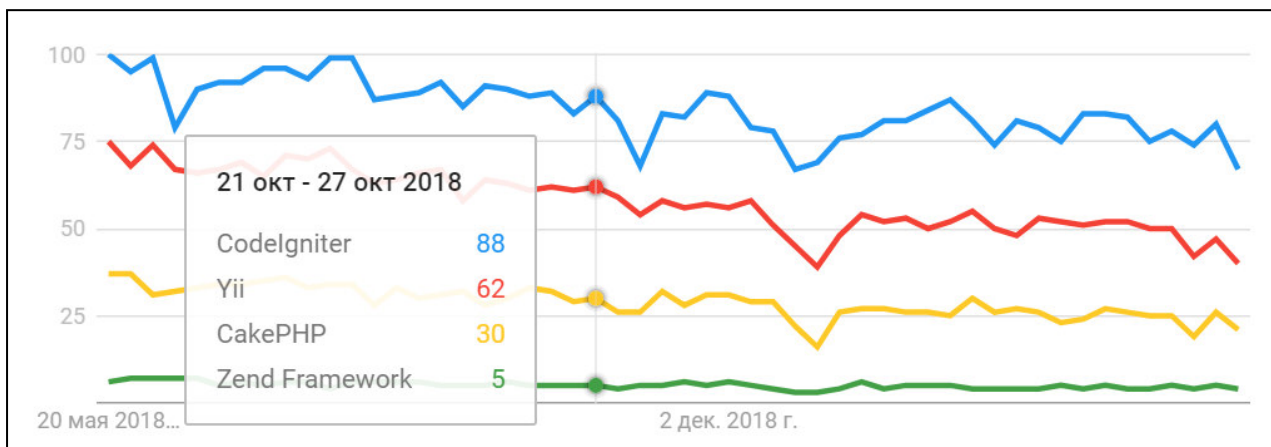
Сонымен қатар, php-фреймворк yii2 таңдалынды. Yii – ауқымды веб-қосымшаларды әзірлеу үшін өте тиімді фреймворк [11]. Ол кодты қайта пайдалану тұжырымдамасын барынша қолдануға мүмкіндік береді және веб-әзірлеу процесін айтарлықтай тездетеді. Yii атауы қарапайым, тиімді және кеңейтілген дегенді білдіреді.

Сондай-ақ, фреймворк әзірлеушілерінің бірі – ресейлік Александр Макаров. Ол Ресейде фреймворкты насихаттаумен айналысады және бұл фреймворктің тек Ресейде ғана емес ТМД-да және біздің Қазақстанда кең таралуына әсер еткен ең басты факторлардың бірі (2.1-сурет).



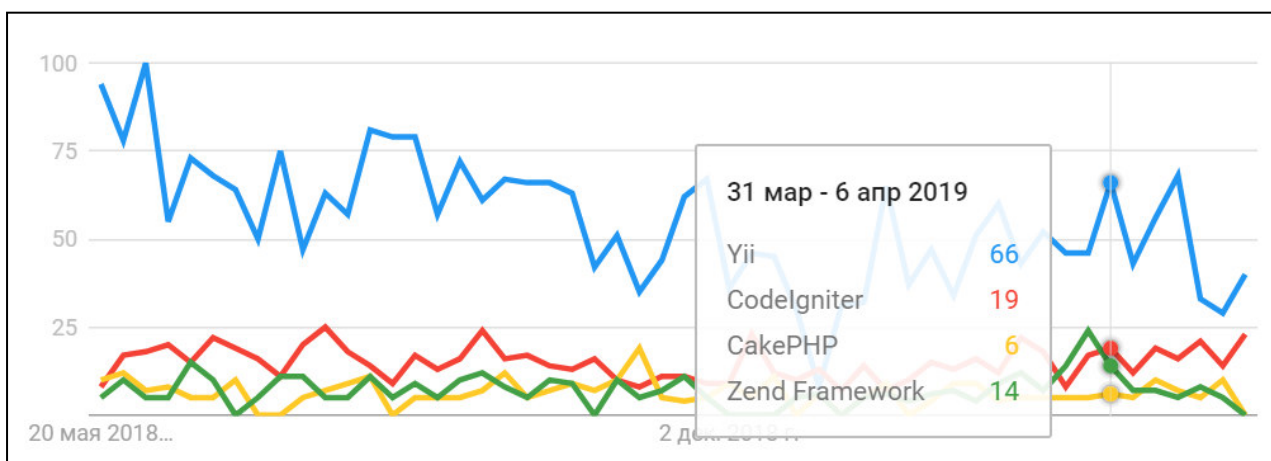
2.1-сурет – Yii 2 фреймворгін қолданатындар

**Yii2 танымалдығы.** Таңдау тым үлкен және тиімді әрі өзіме ыңғайлы фреймворк таңдау өте қиын болды. Көптеген басқа әзірлеушілер сияқты мен де танымал әрі сенімдісін таңдауым керек болғандықтан, Google Trends ұсынатын сұраныс статистикасына жүгіндім [29]. Ең алдымен, әлемдегі фреймворктерді қолдану статистикасын қарастырдым (2.2-сурет).



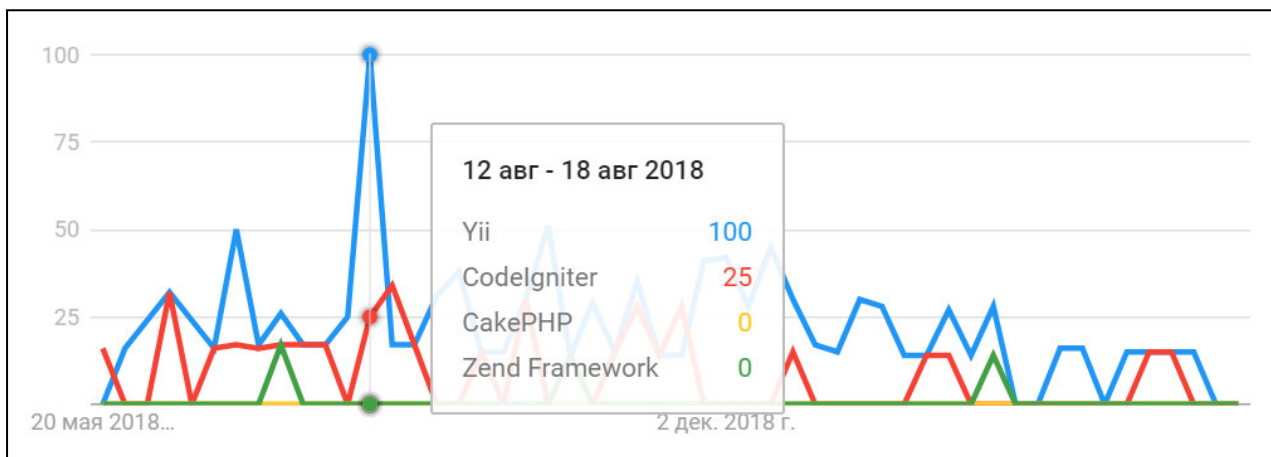
2.2-сурет – Фреймворктерді қолданудың әлем бойынша көрсеткіші

Графиктерден көрініп тұрғандай, Yii фреймворк сенімді екінші орынды алады. Жалпы, бұл жалпы әлемдік статистика екенін ескерсек өте жақсы көрсеткіш. Бірақ егер біз Google Trends-ке Ресейдегі сұраныс статистикасын тандасак, сурет айтарлықтай өзгереді (2.3-сурет).



2.3-сурет – Фреймворктерді қолданудың Ресейдегі көрсеткіші

2.4-суреттен Yii2 фреймворгі Қазақстандағы ең танымал фреймворк екенін көруге болады.



2.4-сурет – Фреймворкті қолданудың Қазақстандағы көрсеткіші

**Yii2 мүмкіндіктері.** Фреймворк танымалдығы бос жерде пайда болмайды. Фреймворк осындай танымалдыққа ие болу үшін ол ең алдымен өте кең мүмкіндіктерге ие болуы және өте икемді болуы керек. Yii2 мүмкіндіктері:

- Yii2-ең жылдам фреймворктардың бірі;
- зерттеу өте оңай;
- жобаның құрылымы MVC паттерні бойынша құрылады;
- деректер қорымен жұмыс істеу үшін DAO және ActiveRecord кластары бар. DAO — бұл деректер қорына тікелей сұраныс беруге арналған класстар жиынтығы. ActiveRecord — кесте класспен, ал кесте жазбалары - класс даналарымен ұсынылатын ORM тәсілін іске асыратын класс. DAO да, ActiveRecord та, PDO арқылы жұмыс істейді, осыдан қолдау көрсетілетін деректер қорының үлкен тізімі шығады: SQLite, MySQL/MariaDB, PostgreSQL, SQL Server, Oracle, Firebird/Interbase 6 және басқалар;
- кіріктірілген интернационалдандыру қолдауы бар;
- PHP базалық кодын Gii генераторымен генерациялау, ол бастапқыда кіріктірілген Yii модулі болып табылады. Gii генераторы көмегімен модел, контроллер және views-ты генерациялауға болады. Gii-да кодты генерациялау үлгілері бар, яғни оны қалағаныңызша өзгертесіз;
- беттерді және жеке фрагменттерді кештеу;
- рәсімдеу тақырыптарын қолдау;
- бөгде класстарды, кітапханаларды қарапайым қосу мүмкіндігі. Яғни, сізге ұнамаса CRM-ді қосуға болады;
- Yii-ді басқа жобаға өте оңай қосуға болады. Файл қосылады, қолданба данасы жасалады және оны пайдалануға болады;
- деректер қорын миграциялау;
- REST-ті қолдау;
- аутентификация және авторизация. Менің ойымша, барлығы орнымен және түсінікті жүзеге асырылған. Қысқа түсіндіруге тырысамын.



Қолжетімділікті шектеу RBAC рөлдер негізінде құрылады. Рөл-міндет-операция иерархиясы құрылады. Және ағаш құрылымы үлкен болуы мүмкін. Рөл пайдаланушыға беріледі. Пайдаланушы бірнеше рөлге ие болуы мүмкін. Осы иерархия бойынша операция, тапсырма және пайдаланушы арасындағы байланыстың бар-жоғын тексеруге болады. Осылайша, пайдаланушы үшін осы әрекеттің қол жетімділігін тексеру. Менің құрған жүйемде админ, менеджер және түлек, яғни үш рөл бердім. Админге барлық операцияға рұқсат жасасам, менеджер тек қана құжатты жібере алады, ал түлек сол құжатты сақтай алады және өзінің суретін өзгерте алады. Тағы бір ерекшелік иерархия базада да, файлда да және басқа да түрде сақталуы мүмкін. Авторизация контроллерге сүзгі түрінде орындалған. Үйі сүзгісі - бұл контроллердің іс-әрекетін орындау алдында орындалатын және ағымдағы жағдайларда контроллердің іс-әрекетін орындау мүмкін еместігін шешетін код;

- кателерді ұстап қалу және өңдеу;
- автоматты тестілеу;
- функционалды кеңейту бойынша үлкен мүмкіндіктер бар - компоненттер мен мінез-құлық, модульдер мен т. б. және т. б.;
- сондай-ақ қауіпсіздік те деңгейі де жоғары. Cookie арқылы шабуылдардың алдын алу, cookie қолдан жасалған сұрауларды болдырмау үшін мүмкіндіктер жасалған.

Қорытындылай келе Үйі2-жақсы қоғамдастық және құжаттары орыс тілді PHP-дегі динамикалық дамып келе жатқан фреймворк. Жақсы қоғамдастық және құжаттама оны зерттеу және пайдалану үшін жеткілікті ыңғайлы және қол жетімді етеді. Фреймворк танымалдығының арқасында, ол үшін көптеген дайын жазулар бар. Үйі2-де оның функционалдық мүмкіндіктерін кеңейту үшін бөгде кітапханаларды пайдалану ыңғайлы болып табылады. Тіпті WordPress, Joomla сияқты бөгде жүйелерді әзірлеу және жақсарту үшін де кітапхана ретінде пайдалану мүмкіндігі бар.

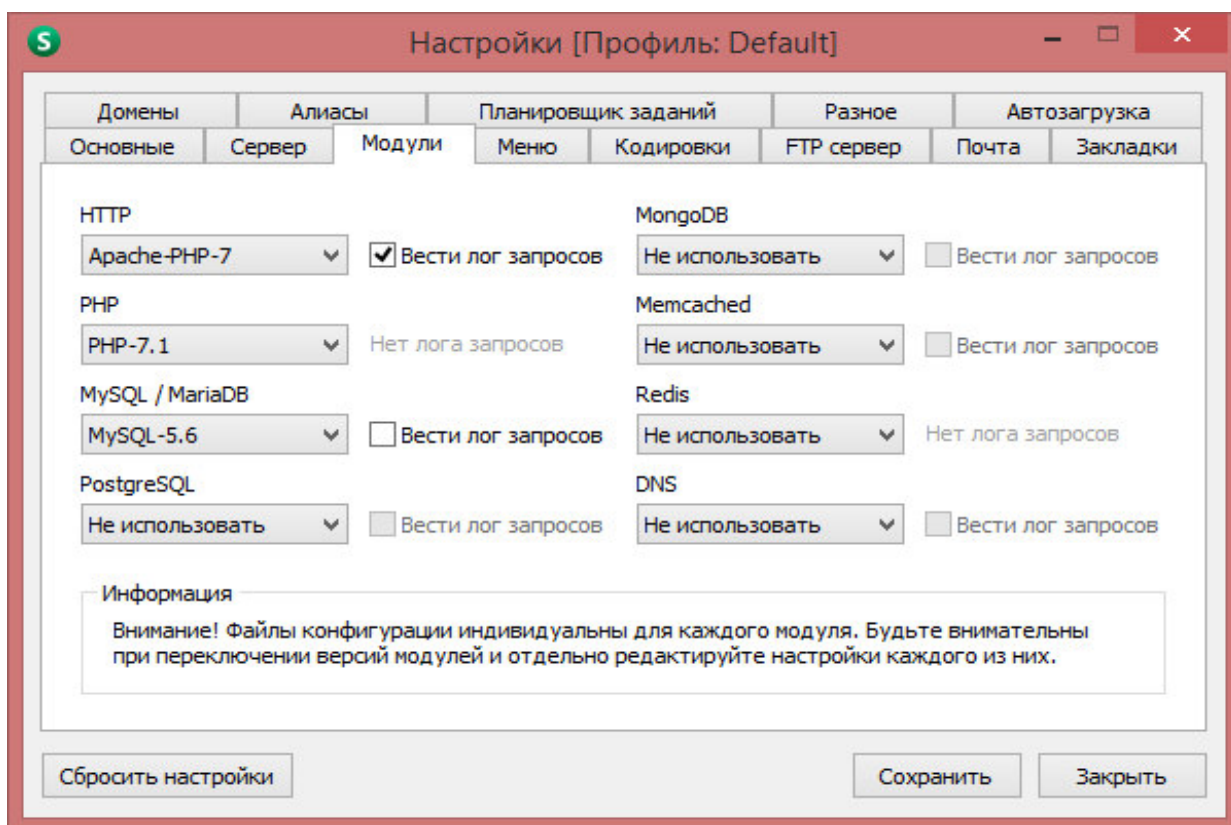
Үйі2 қосымшасымен жұмыс істеу үшін PHP тілі, мәліметтерді басқару жүйесі – MySQL және веб-сервер, менің жағдайымда OpenServer, орнатылуы керек. Сондай-ақ Үйі2 қосымшасының талабына сай PHP тілінің 5.4 нұсқасынан жоғары нұсқасын таңдау керек (2.5-сурет).

Open Server Panel - бұл веб-әзірлеушілер үшін арнайы жасалған ұсыныстар мен ұсыныстар негізінде құрылған портативті сервер платформасы және программалық жасақтама ортасы [30].

Программалық кешен серверлік программалық қамтамасыз етудің бай жиынтығына, ыңғайлы, көп функциялы, жақсы ойластырылған интерфейске ие және компоненттерді басқару және теңшеу үшін қуатты мүмкіндіктерге ие. Платформа веб-жобаларды әзірлеу, түзету және тестілеу, сондай-ақ жергілікті желілерде веб-қызметтерді ұсыну үшін кеңінен қолданылады.

Алғашында комплекске қосылған программалық өнімдер бір-бірімен жұмыс істеуге арналмағанымен, мұндай пакет Windows пайдаланушылары арасында өте танымал болды, бірінші кезекте олар Linux серверлерінің

деңгейінде сенімділікпен тегін программалық жасақтама жинағын алғандығына байланысты.



2.5-сурет – OpenServer қосымшасының баптаулары

Ыңғайлылық пен басқарудың ыңғайлығы, әрине, сізді бей-жай қалдырмайтын болады, ол болған кезде Open Server әрбір вебмастерге қажетті бірінші класты және сенімді құрал ретінде өзін танытты.

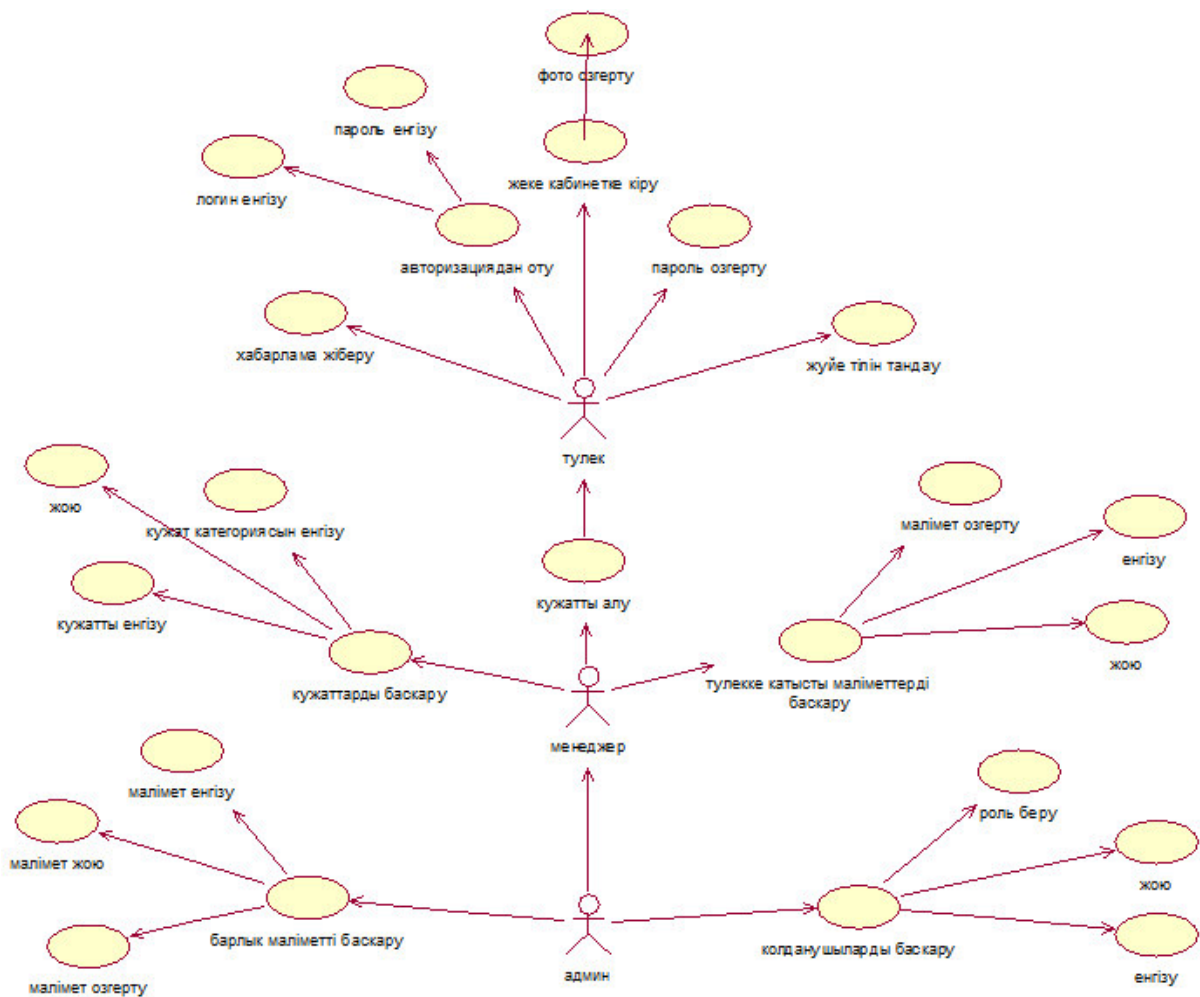
OpenServer басқару программасының ерекшеліктері:

- Windows науасында көрінбейтін жұмыс;
- жылдам іске қосу және тоқтату;
- программаны іске қосқанда autostart сервері;
- бірнеше доменді басқару режимдері;
- виртуалды дискіні орнату;
- пәрмен жолымен қолдауды басқару;
- профильдер параметрлерін қолдау;
- барлық компоненттердің ыңғайлы қарау журналдары;
- HTTP, MySQL және PHP модульдерін ауыстыру;
- толық және түсінікті құжаттама;
- домендерге бір рет басу арқылы кіру;
- конфигурациялық үлгілерге тез қол жеткізу;
- көптілді интерфейс;
- тізімдегі авторлық программалар.

## 2.2 Программаны әзірлеу құралдары

«Университет түлектерінің бірлестігі» веб-бағытталған жүйесі үш жақты қолданушыларға негізделген: админ, менеджер және түлек.

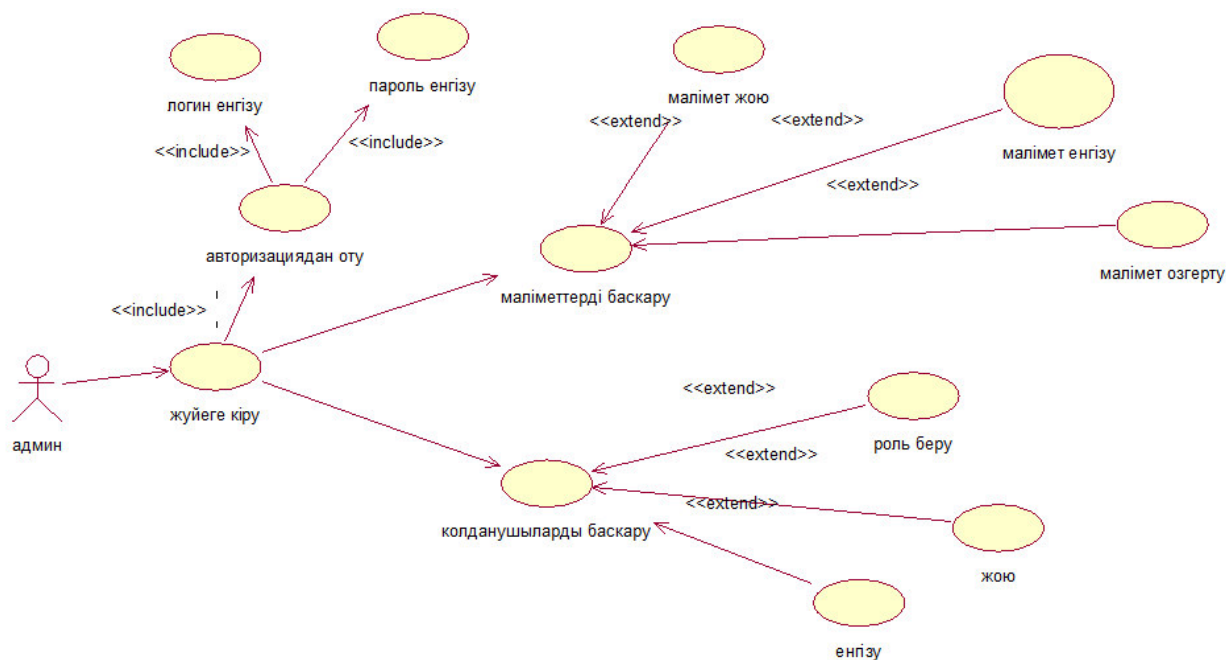
2.6-суретте көрсетілген диаграмма қолданушылар орындай алатын іс-әрекеттерді көрсетіп тұр, яғни админ барлық іс-әрекеттерді басқарады, соның ішінде қолданушыларды, мәліметтерді, құжаттарды және т.б. Ал менеджер болса, түлекке қатысты мәліметтер мен құжаттарды ғана өзгерте алады және түлек секілді жеке кабинетке кіреді, пароль өзгертеді, хабарлама жібереді және т.с.с.



2.6-сурет – Жүйенің қолдану нұсқаларының жалпыланған диаграммасы

Админ – пайдаланушының рөлі. Сондай-ақ, админ басқа пайдаланушыларға сайт админі рөлін тағайындай немесе жою алады (2.7-сурет). Админ рөлін басқа пайдаланушыларға тағайындағанда, көрсетілген рөлмен сіз оларды сайттан мазмұнды алып тастау мүмкіндігіңіз бар екенін есте сақтау өте маңызды, сондықтан өте сақ болу керек.

Тек админ сайтты жаңарта алады. Мысалы, ол тақырыптарды өзгертіп, негізгі файлдарды кірістірілген тақырып редакторымен өңдей алады. Мұнда админ сайтта кез келген плагиндерді қосуға, жоюға және өзгертуге құқылы.



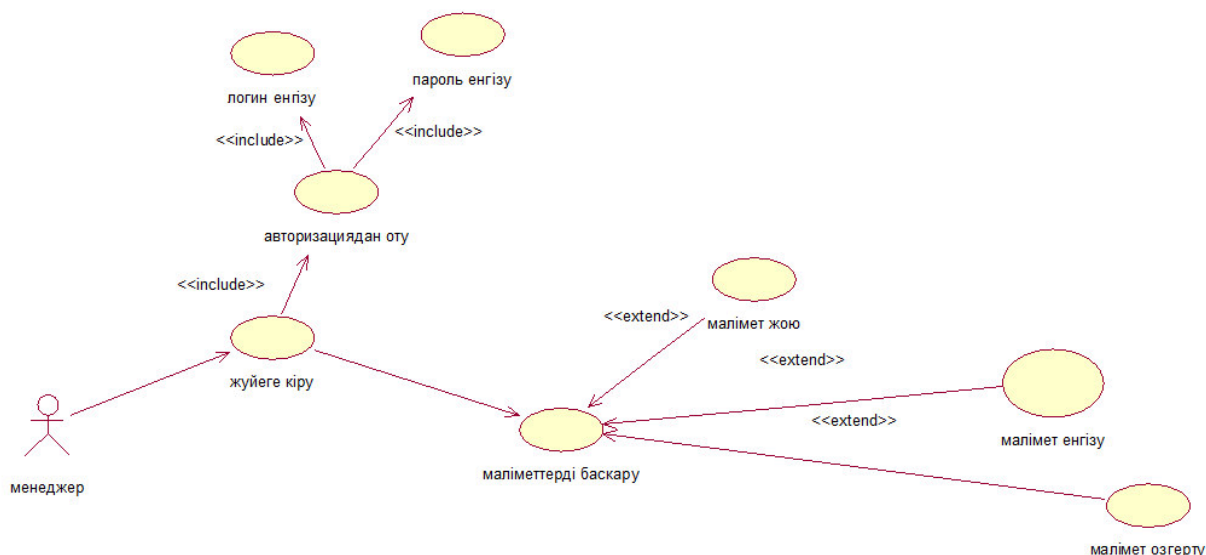
2.7-сурет – Админ диаграммасы

Сайтты баптармен толтыру ақпараттың сапасы, оның оқу және қабылдау жеңілдігі, өзектілігі және пайдаланушының сұрау салуына сәйкестігі маңызды болып табылады.

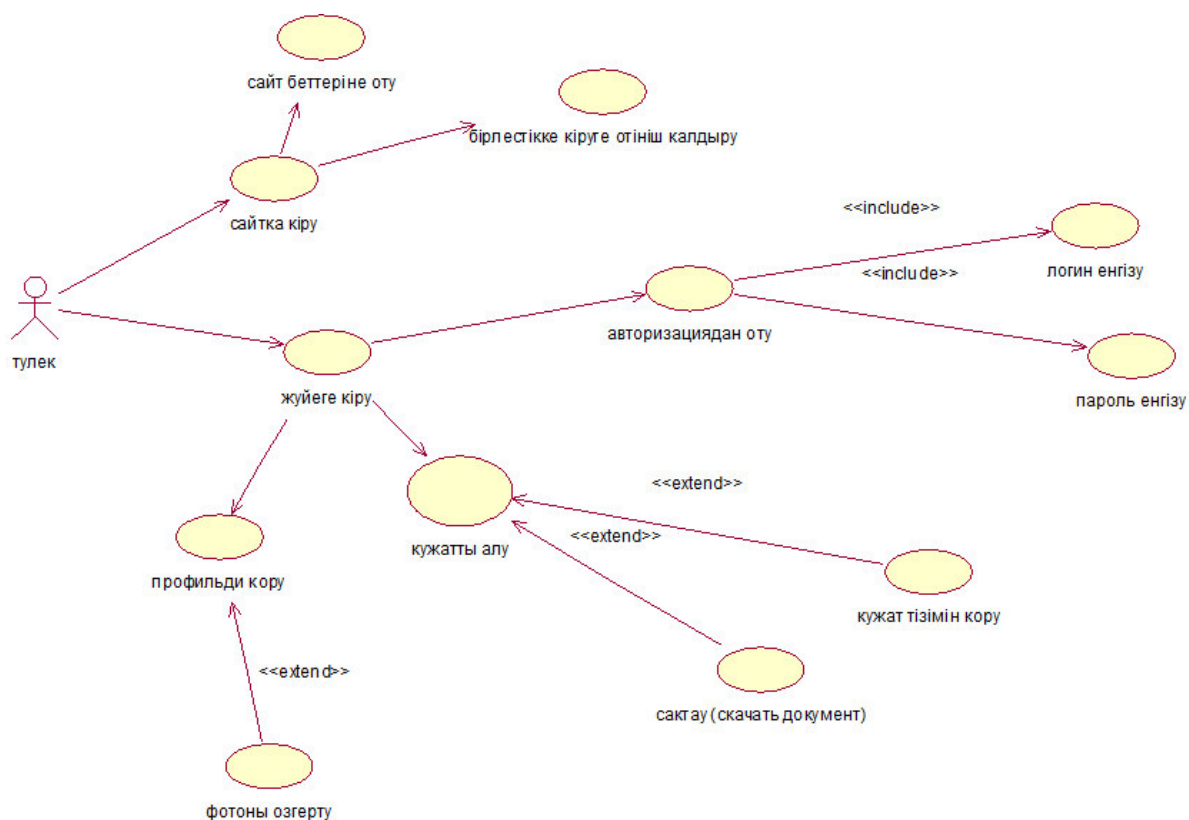
Әр түрлі сайттарға әртүрлі ақпараттық мазмұн талап етіледі, сондықтан менеджер сайттың ерекшеліктерін түсінуі тиіс.

Менеджер өзіне сеніп тапсырылған сайтта басқаруды толық меңгеруі, оны өзекті және жұмысқа қабілетті күйде ұстауы тиіс.

Бұл жүйедегі менеджер мәліметтерді ғана басқарып қоймай, түлектерге хабарламалар, шақыртулар жән ескертпелер жіберіп отырады. Кез келген кафедра немесе деканат қызметкері бұл ролге ие бола алады, админ арқылы жүйеге тіркеледі. Менеджер іс-әрекеті 2.8-суретте көрсетілген.



2.9-сурет – Менеджер диаграммасы

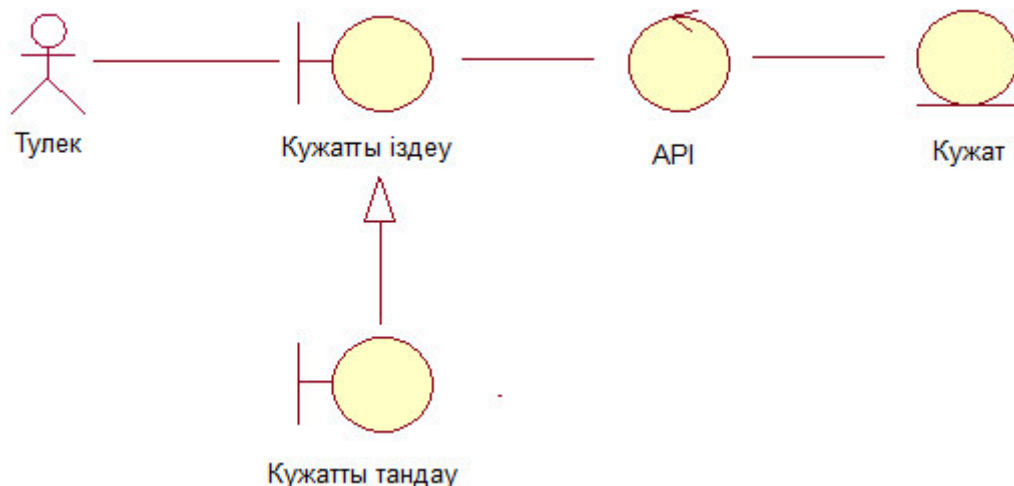


2.8-сурет – Түлек диаграммасы

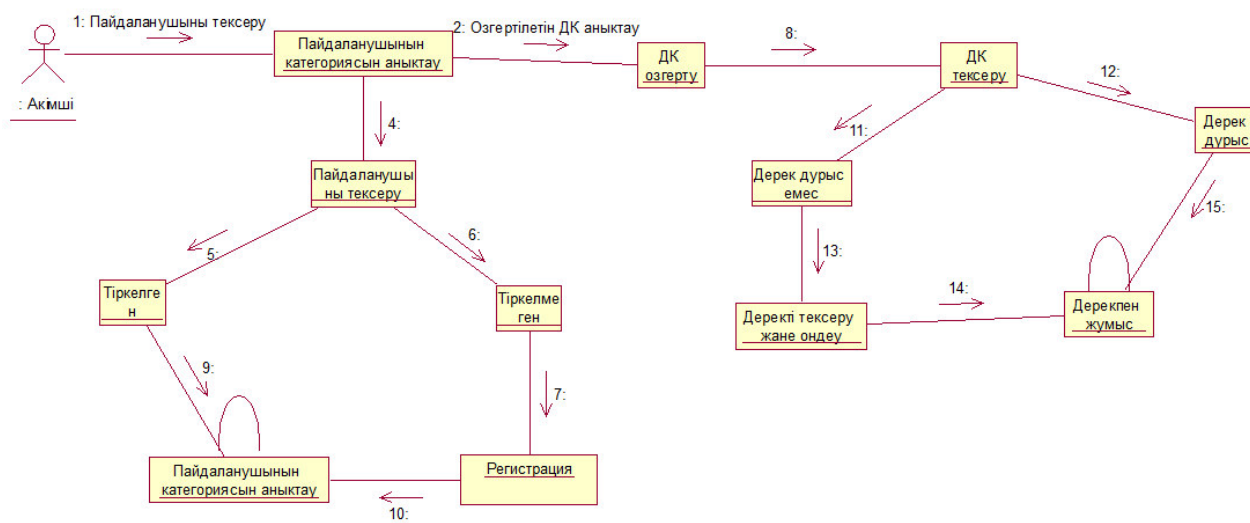
Түлектке қажетті құжат, анықтамаларды менеджер жеке кабинетке салады, процесс қалай жүретіні 2.9-суретте көрсетілген. Сәйкесінше түлек сол құжаттарды өз жеке кабинетіне кіру арқылы көре алады және өзіне тиесілі әрекеттерді жасайды (2.10-сурет).



2.9-сурет – «Кұжатты салу» анализ диаграммасы

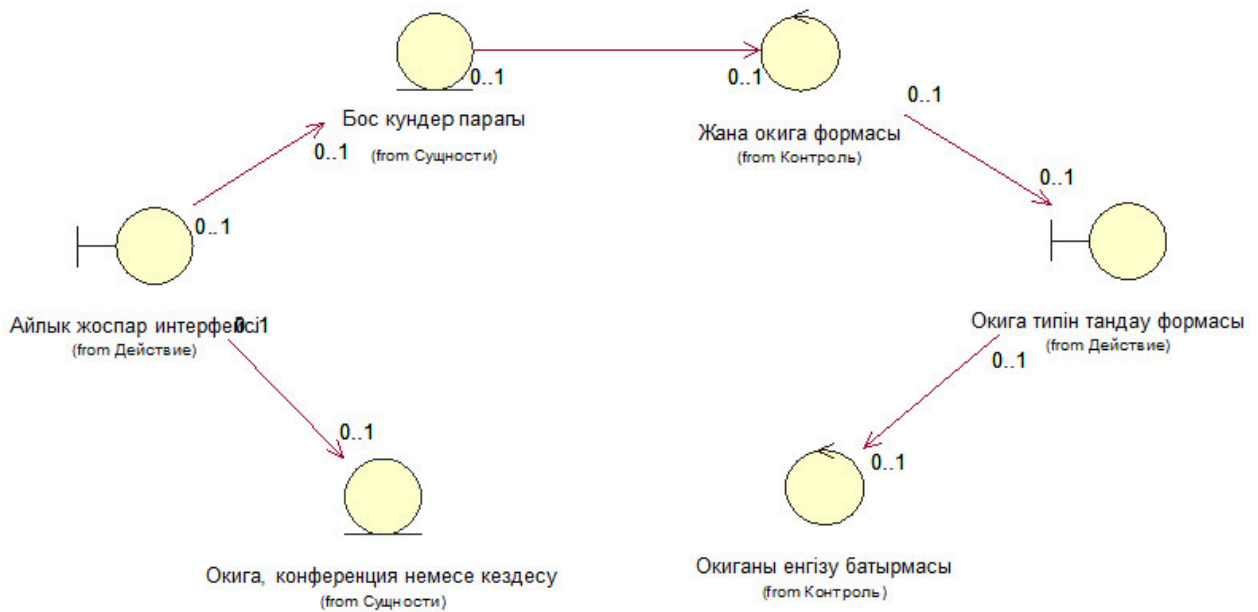


2.10-сурет – «Кұжатты алу» анализ диаграммасы



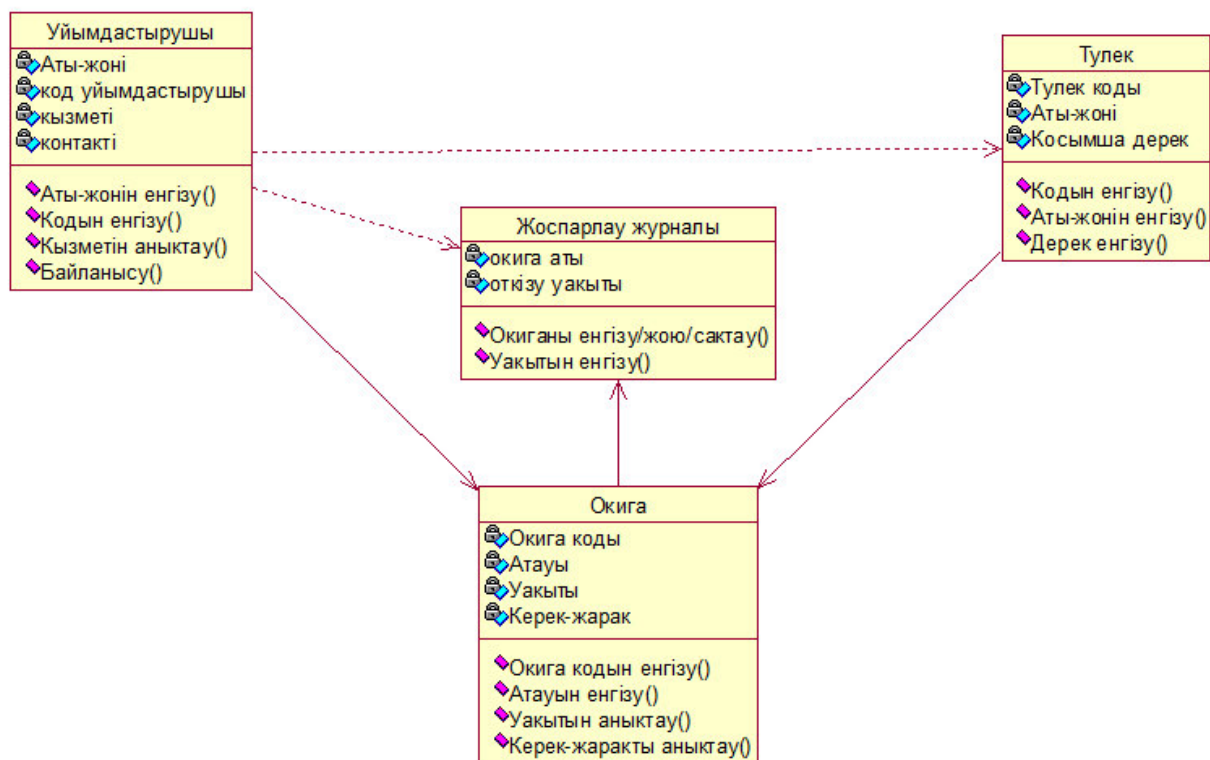
2.11-сурет – Логикалық диаграмма

Логикалық жоспарлау диаграммасына мысал ретінде оқиға енгізуді қарастырдым (2.12-сурет). Түлектерді кездесуге немесе іс-шараларға шақыруға болады, ол админ және менеджер арқылы жүзеге асырылады. Яғни, оқиға уақыты, өтетін орны туралы мәліметтер енгізіп, қолданушы типіне түлектер деп белгілейміз де жібереміз. Түлек өз жеке кабинетіне кірген кезде енгізілген оқиға көрінеді.



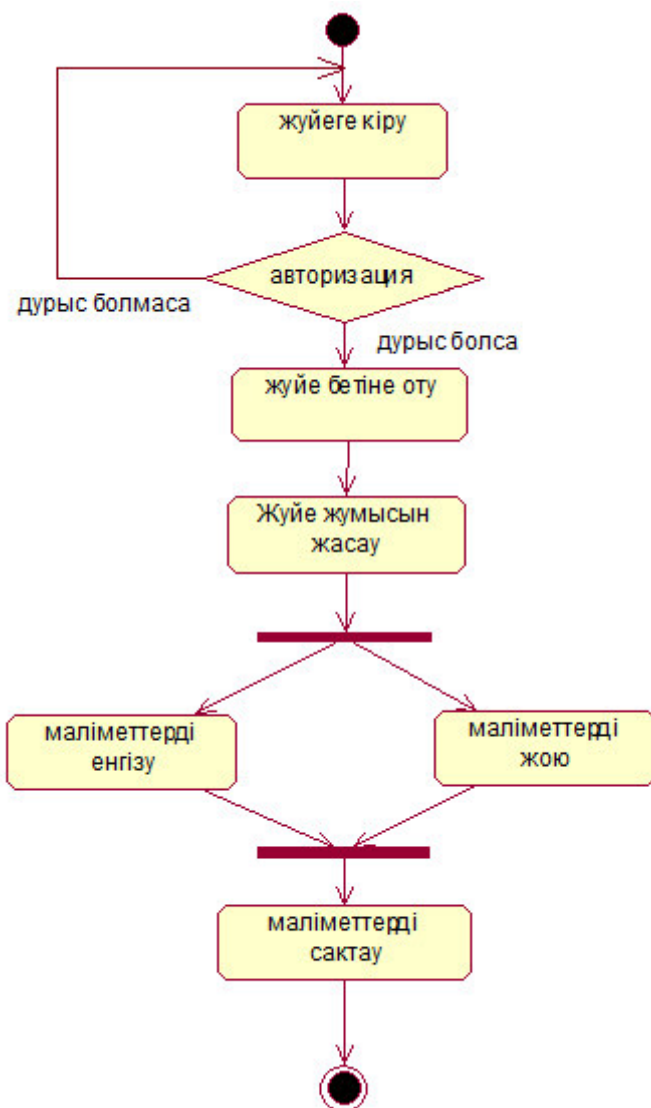
2.12-сурет – Оқиға енгізудің логикалық жоспарлау диаграммасы

UML тіліндегі класс басқа құрылымдардағы объектілермен бірдей құрылымға, мінез-құлыққа және қарым-қатынастарға ие объектілер жиынтығын белгілеу үшін пайдаланылады. Оқиға енгізудің класстар диаграммасы 2.13-суретте көрсетілген.



2.13-сурет – Оқиға енгізудің класстар диаграммасы

Процесс бастапқы нүктеден басталады, содан кейін "Жүйеге кіру" күйіне ең бірінші өту керек. Жүйедегі объектінің мінез-құлқына ауысулар мен күйлер көрсететін әрекеттерді жатқызуға болады. Басқа болса да, бұл "Жүйе" класының кейбір әдісімен жүзеге асырылатын процестер әртүрлі жолмен түсіндіріледі. Іс-қимылдар ауысумен байланысты және жылдам әрі еңсерілмейтін ретінде қарастырылады. Қызмет жағдайларға байланысты жеткілікті ұзақ уақытқа созылуы мүмкін немесе белгілі бір оқиғаға байланысты әрекетті тоқтатуға болады.



3.14-сурет – Жүйе админінің қызмет диаграммасы

Осылайша, объектілі-бағытталған талдау және күрделі жүйелерді жобалау процесі қызметтің диаграммасын қоса алғанда, жекелеген диаграммаларды төмендеу және өрлемелі әзірлеудің бірізділігі ретінде ұсынылады. Әзірлеудің қандай да бір бағыттарының Үстемдігі нақты жобаның ерекшеліктерімен және оның жаңалығымен айқындалады.



### 3 Программалық интерфейс

Сайттың интерфейсі қандай болуы керек?

**Бірегей дизайн.** Барлық беттер үшін бірыңғай стильде жасалынып, сайт тақырыбына жауап беру керек. Сайттың түс гаммасы қолданушыларға жағымды болуы керек. Графика мен дизайн есте қаларлық, барлығы қарапайым және қолжетімді болуы керек.

**Оңай навигация.** Сайттың барлық элементтері, мәзірлері және басқа да ақпараты ыңғайлы орналастырылуы тиіс, себебі келуші дереу бетке бағытталып, қажетті ақпаратты таба алады.

**Түстер дизайны.** Бұл ерекше көзқарасты талап ететін маңызды мәселе. Әрбір адамға арналған барлық түсті әртүрлі әсер етеді, біреудің қызғылт түсті ұнатпауы мүмкін, біреу оны ұнатуы мүмкін. Сондықтан сайтты бейтарап түстерде жасау ұтымды.

**Жылдам бет жүктемесі.** Яғни, сайт жылдамырақ жүктелуі үшін беттердің мөлшері мүмкіндігінше аз болуы керек. Бұл баннерлерге, сайтта анимацияланған элементтерге, шағын, қайталанатын, жарнамаларға және т.б. қатысты.

**Тұрақты немесе резеңке дизайн.** Күрделі дизайн жақсы көрінуі керек және түрлі экрандық ажыратылымдарда дұрыс көрсетілуі тиіс.

Қазіргі заманғы интерфейстер:

- командалық интерфейс;
- WIMP — интерфейс;
- SILK интерфейсі.

Командалық интерфейс деп аталады себебі, адам компьютерге команда береді, соған сәйкес компьютер оны орындайды және нәтижесін шығарады. Командалық интерфейс бейнепакеттік технология мен командалық жолдың технологиясында іске асырылған.

WIMP – интерфейсте (Window — терезе, Image — сурет, Menu — мәзір, Pointer-көрсеткіш) диалог команда көмегімен емес графикалық бейнелердің – меню, терезе, басқа да элементтер көмегімен жүзеге асады. Бұл интерфейсте де машинаға командалар беріледі, бірақ графикалық бейнелер арқылы "жанама" жасалады.

SILK интерфейсі (Speech – сөйлеу, Image – сурет, Language – тіл, Knowledge – білім) видинтерфейс қарым-қатынастың қарапайым, адами түріне ең жақын. Қазіргі уақытта, бұл жұмыс өте қиын. Бұл компьютер адам тілін талдай отырып, негізгі фразаларды таба отырып, өзі үшін командаларды табады. Командаларды орындау нәтижесі адамға түсінікті пішінге айналады. Бұл интерфейстің түрі компьютердің аппараттық ресурстарына аса талап етілген, сондықтан оны негізінен әскери мақсаттар үшін қолданады.

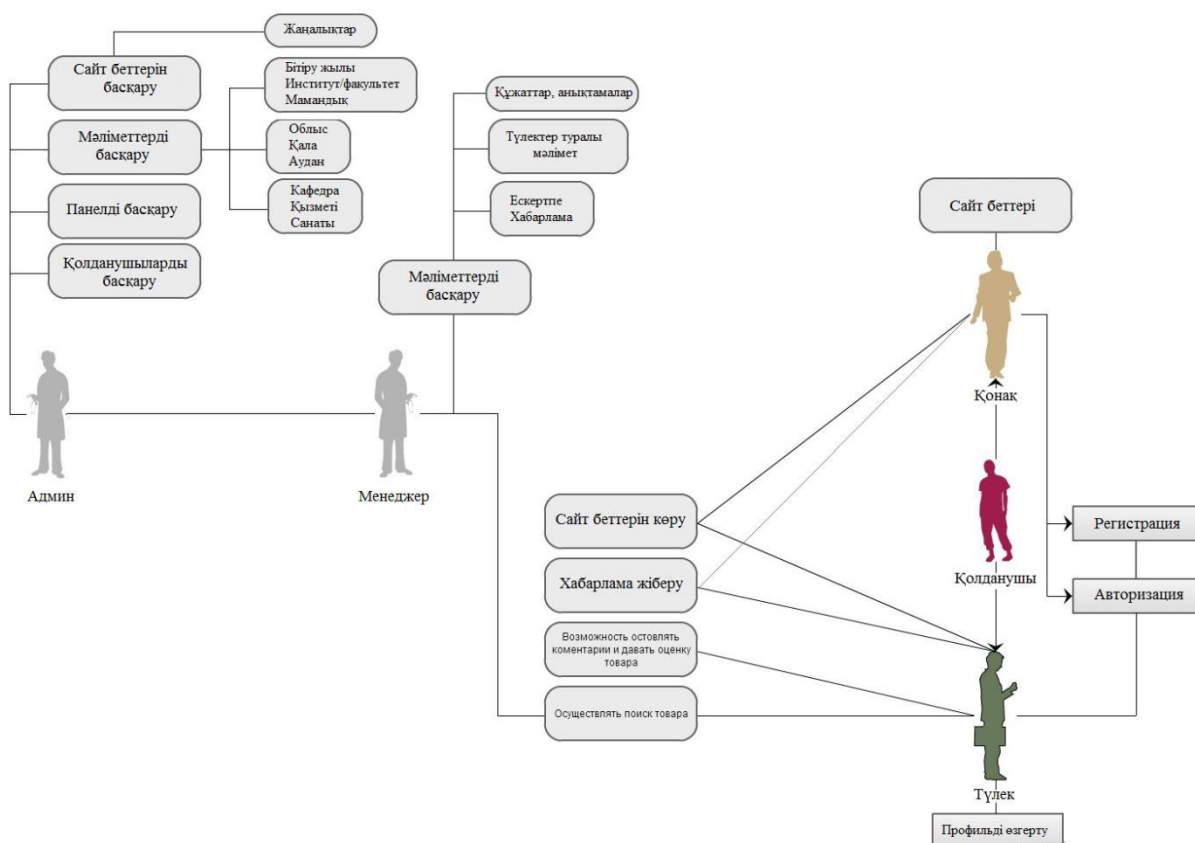
Сайт құрылымының әрбір бөлімі сілтеме арқылы ақпарат алуға мүмкіндік береді (3.1-сурет). Жүйе үш жақты қолданушыларға негізделген: админ, менеджер және түлек. Админің міндеттері:

- пайдаланушыларға арнап мәліметтерді енгізу және оларды өзгерту, оның ішіне менеджерлер мен түлектер кіреді;
- пайдаланушылардан келген хабарламаларға жауап беру;
- пайдаланушыларға рөл, яғни рұқсат беру;
- өтініш жіберген түлек АЭЖБУ-да оқығандығы расталса, жүйеге енгізу;
- жаңалықтарды өзгерту;
- сайт беттерін өзгерту.

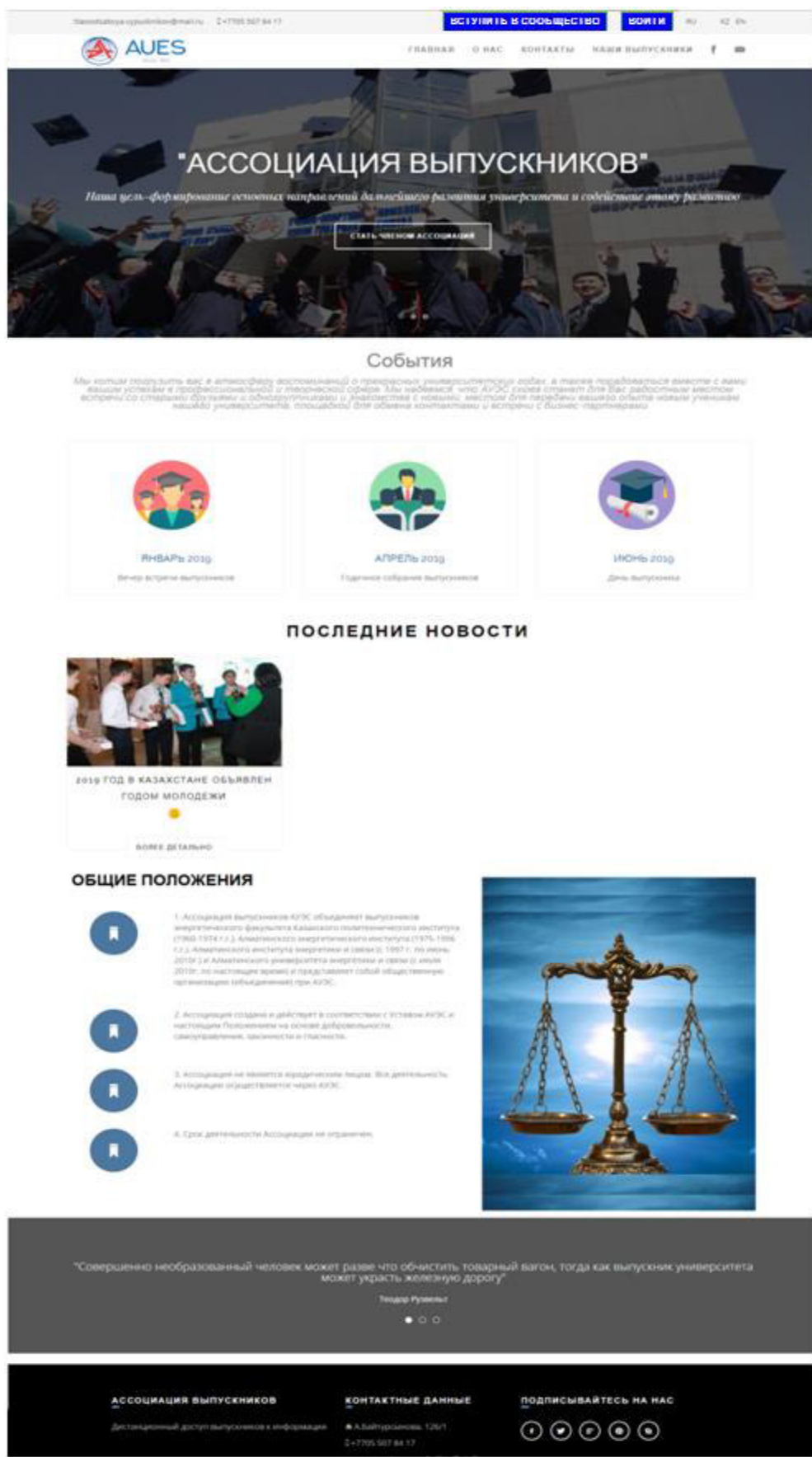


3.1-сурет – Сайттың логикалық құрылымы

Админ, менеджер және түлекке арналған жүйенің жұмыс жасау көрінісі 3.2-суретте көрсетілген.



3.2-сурет – Жүйенің жұмыс жасау көрінісі



3.3 -сурет – «Университет түлектерінің бірлестігі» веб-бағытталған жүйесінің алғашқы көрінісі

## О НАС

### ОБ АССОЦИАЦИИ

Членами Ассоциации могут быть выпускники АУЭС, а также энергетического факультета Казахского политехнического института (1960-1974 г.г.) и Алматинского энергетического института (1975-1996г.г.), Алматинского института энергетики и связи (с 1997г. по июнь 2010г.), признающие настоящее Положение, принимающие участие в работе Ассоциации и уплачивающие членские взносы. В Ассоциацию могут входить наряду с физическими лицами (в т.ч. выпускники других вузов идентичных специальностей университета) и юридические лица (как правило, профильные предприятия энергетики, телекоммуникации, IT-технологии). Ежегодный членский взнос вносится членами Ассоциации в размере, определяемом Советом выпускников-попечителей АУЭС. Денежные средства Ассоциации образуются из членских взносов и добровольных спонсорских пожертвований, аккумулируемых в Фонде Ассоциации на расчетном счете АУЭС. Фонд Ассоциации выпускников имеет субсчета выпускающих кафедр и других подразделений университета с правом расходования аккумулируемых на каждом субсчете средств Советом выпускников-попечителей на нужды соответствующей кафедры или подразделения университета, или по их предложению – на общеуниверситетские нужды.



### ЧАСТО ЗАДАВАЕМЫЕ ВОПРОСЫ:



#### ЧТО ТАКОЕ АССОЦИАЦИЯ ВЫПУСКНИКОВ АУЭС?

Ассоциация выпускников АУЭС – это деловой и социальный клуб, среди главных задач которого – предоставить выпускникам возможности для общения между собой и достижения собственных целей. Наше сообщество создано для того, чтобы наладить плодотворное общение университета с его выпускниками, а также выпускников и студентов между собой.



#### ДЛЯ ЧЕГО СОЗДАНА АССОЦИАЦИЯ ВЫПУСКНИКОВ АУЭС?

Ассоциация создана для тех, кто интересуется вопросом: "Что такое выпускники АУЭС, зачем нужна ассоциация выпускников?" а также для

## 3.4-сурет – «Біз туралы» беті

### КОНТАКТЫ

**Ауэс корпус А**  
Виз  
★ ★ ★ ★ ★ 4.9  
☎ +7 (727) 292-03-03  
🌐 aues.kz  
ул. Байтурсынова, 126, Алматы, Казахстан  
• Байконур — 740 м Путь от метро  
• Театр им. Ауэзова — 1,2 км Путь от метро  
• АБЖ — 1,7 км Путь от метро

Как добраться Об организации

#### НАШ АДРЕС

Республика Казахстан  
г. Алматы  
А.Байтурсынова, 126/1  
Тел: +7795 587 84 17  
Тел: +7747 804 74 55  
Email: assoatsatsiya.vypusknikov@mail.ru

Если есть вопросы или предложения свяжитесь с нами:

Имя... Email...

Номер телефона...

Сообщение...

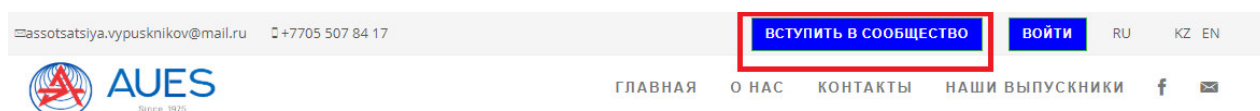
**ОТПРАВИТЬ**

## 3.5-сурет – «Контакті» беті



### 3.6-сурет – «Біздің түлектер» беті

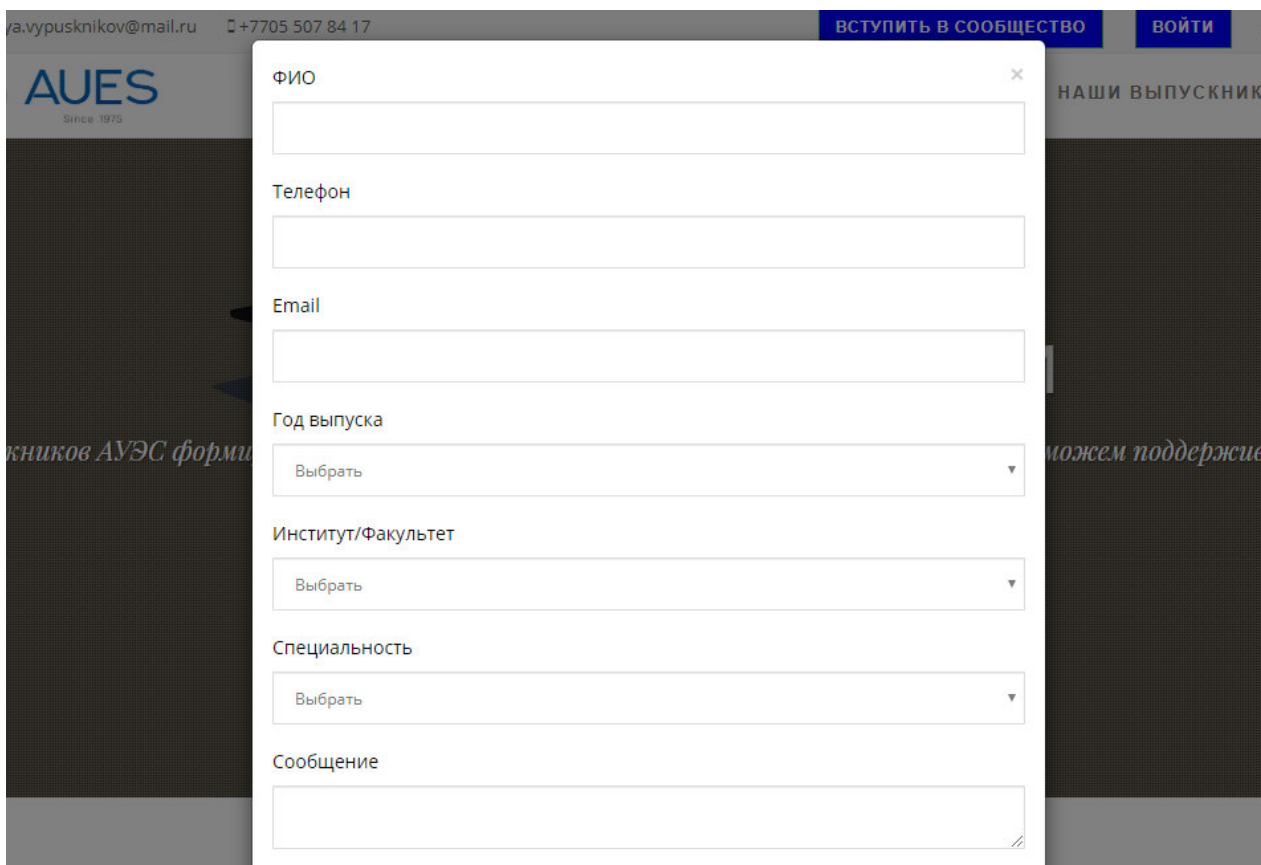
Сайттың оң жақ шетінде «Бірлестікке кіру» батырмасы орналасқан. Ол түлектердің бірлестікке кіруге өтініш тастауына мүмкіндік береді. Оның орналасқан орны 3.7-суретте көрсетілген.



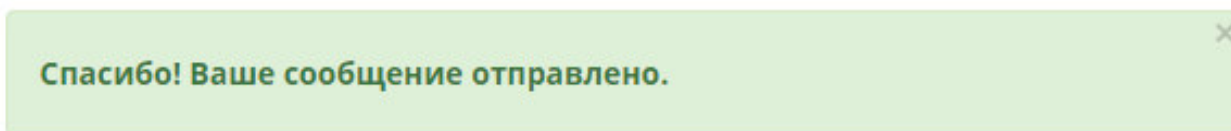
### 3.7-сурет – «Бірлестікке кіру» формасының орналасқан орны мен көрінісі

«Бірлестікке кіру» батырмасы РНР тілінде форма ретінде жасалған және батырманы басқанда сол бетте қалқыған терезе пайда болады (3.8-сурет). Жіберу батырмасын басқан кезде жауап беріледі (3.9-сурет).

Ал жіберілген хабарламалар админ электронды почтасына келіп түседі. Бұл админ почтасын енгізу көрсетілген жолдағы программа кодында көрсетілген: /config/params.php (3.10-сурет). Орталықтың почтасына кіруге кімде рұқсат бар, сол қызметкерлер келген хабарламаларды оқып жауап қайтара алады.



3.8-сурет – «Бірлестікке кіру» батырмасын басқаннан кейінгі көрінісі

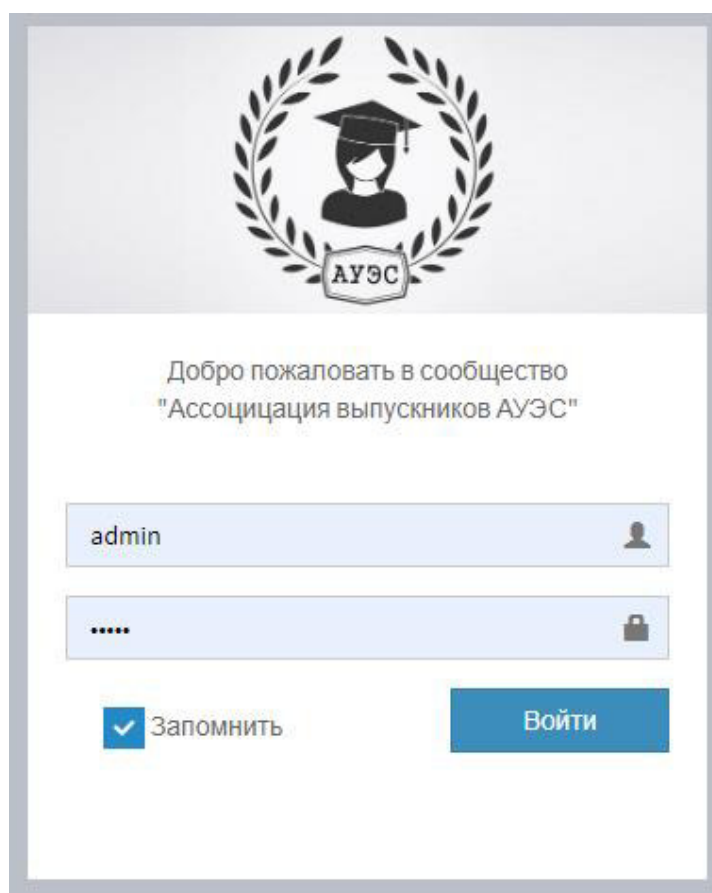


3.9-сурет – Хабарлама жіберілгеннен кейінгі алынатын жауап

```
1 <?php
2
3 ▼ return [
4     'adminEmail' => 'assotsatsiya.vypusnikov@mail.ru',
5 ];
6
```

3.10-сурет – Админ почтасы

Админ бөліміне кіру үшін «Кіру» батырмасын басамыз және ол арнайы логин және құпия сөзді енгізу арқылы орындалады (3.11-сурет). Админ құпия сөзі деректер қорында шифрланған түрде сақталған (3.12-сурет). Егер құпия сөз деректер қорындағы құпия сөзбен сәйкес келмесе, онда құпия сөздің дұрыс еместігі туралы хабар беріліп, қайта енгізу мүмкіндігі беріледі.



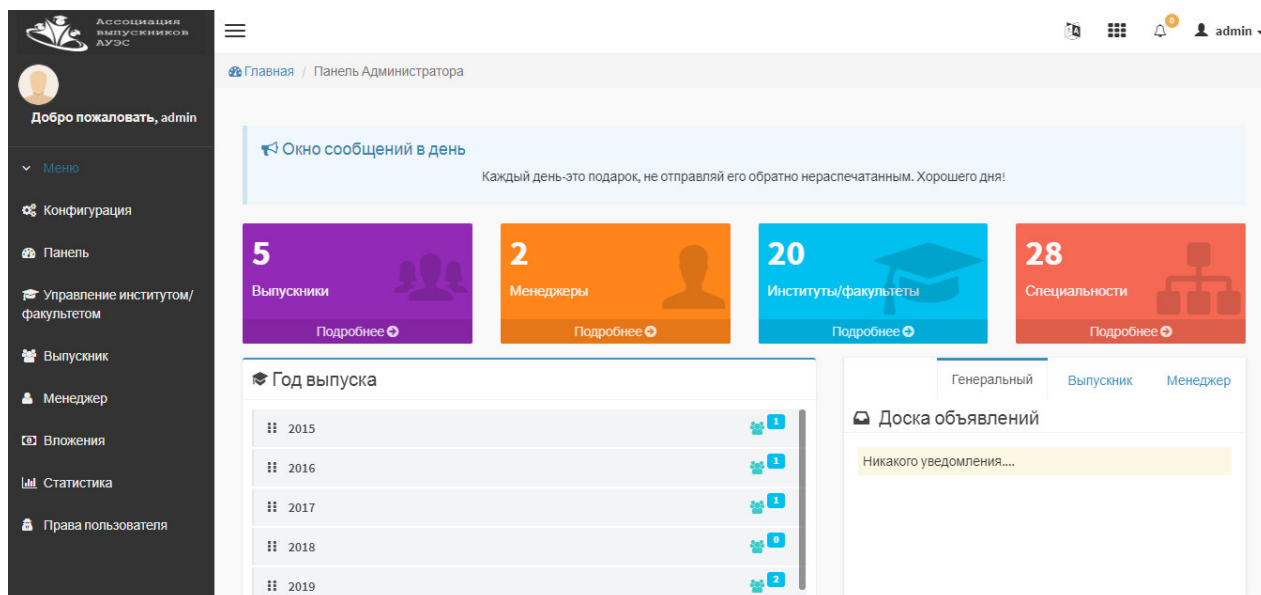
3.11 -сурет – Жүйеге кіру

user_id	user_login_id	user_password	user_type
1	admin	f6fdffe48c908deb0f4c3bd36c032e72	A

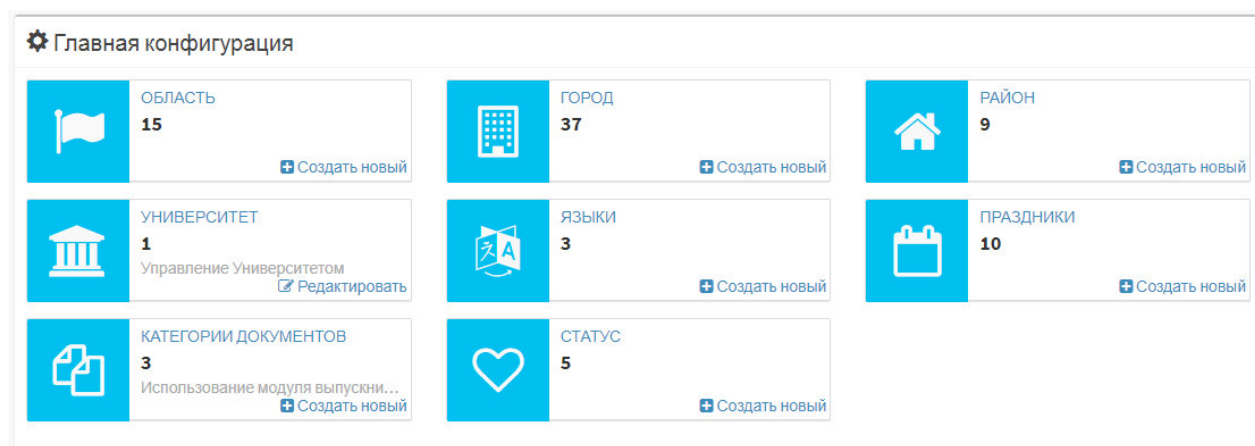
3.12-сурет – Деректер қорындағы админ логині мен паролі

Админ логині және құпия сөзін енгізгеннен кейін, ол автоматты түрде админ бөліміне енеді және мәліметтерді реттейді. Админ бөлімі конфигурация, панель, институт/факультет, мамандықты басқару, түлектер, жинақ, статистика, қолданушылар құқығы бөлімдерінен тұрады. (3.13-сурет).

Конфигурация бөлімінде Университет, мекен-жай, мерекелер, статус жөнінде ақпарат орналасқан, оны админ басқарып отырады (3.14-сурет). Сондай-ақ, түлекке қажетті құжаттар санатын енгізуге болады, сол құжаттардың санатына қарай түлектер өзіне қажетті анықтамаларды өзінің жеке кабинетінен сақтай алады. Облыс, қала, аудан, тілдер, статус түлек немесе менеджерді қосқан кезде қажетті ақпарат болып табылады.



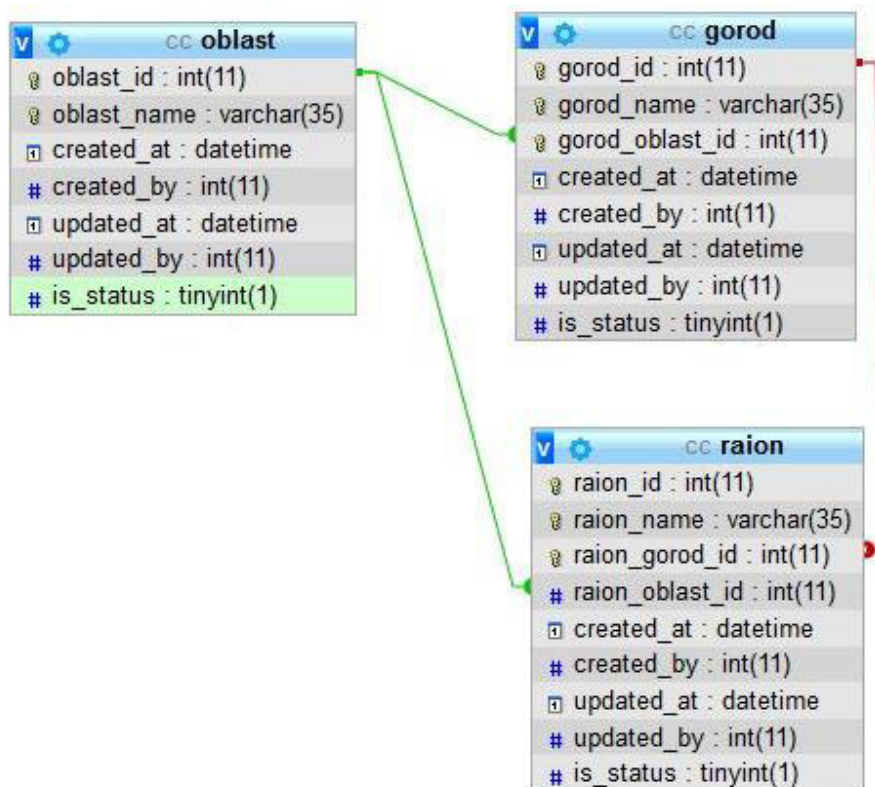
3.13-сурет – «Админ панелі»



3.14-сурет – «Конфигурациялар»

Облыс, аудан, қала сыртқы кілт арқылы байланысқан, себебі мекен-жай туралы мәлімет енгізген жағдайда олар бір-бірімен байланыста болуы керек (3.15-сурет). Мысалы, 3.16-суретте көрсетілгендей қала енгізу үшін облысты таңдау керек.





3.15-сурет – Мекен-жайға қатысты байланыс диаграммасы

+ Добавить город

-- Выбрать область --

- Выбрать область --
- Акмолинская область
- Актюбинская область
- Атырауская область
- Алматинская область
- Восточно-Казахстанская область
- Западно-Казахстанская область
- Карагандинская область
- Кызылординская область
- Костанайская область
- Павлодарская область
- Северо-Казахстанская область
- Южно-Казахстанская область
- Мангистауская область
- Жамбылская область
- Город

PDF
EXCEL

#	Область	Город	Действия
1	Город		🔍 🗑️ ✖️
2	Алматы	Город	🔍 🗑️ ✖️
3	Петропавловск	северо-Казахстанская область	🔍 🗑️ ✖️

3.16-сурет – Қаланы енгізу көрінісі

Университет модулінде АЭЖБУ жайында мәліметтер көрсетілген (3.17-сурет) және оны басқарып, жаңартып отыратын админ болып табылады.

Имя	Алматинский университет энергетики и связи
Псевдоним	AУЭС
Год основания	1975
Адрес	ул. А.Байтурсынова, 128/1
Телефон	+7(727) 9099099
Почта	<a href="mailto:info@aes.kz">info@aes.kz</a>
Вебсайт	<a href="http://www.aes.kz">http://www.aes.kz</a>
Префикс входа выпускника	STU
Префикс логина менеджера	EMP
Время создания	27-04-2019 18:15:15
Создал	admin
Время обновления	10-05-2019 01:03:42
Обновил	admin
Логотип	

### 3.17-сурет – «Университет жайлы мәлімет»

Панель бөлімінде әр қолданушының панеліне арнайы хабарламалар, немесе ескертпе қалдыруға болады және күнтізбеге оқиға енгізу арқылы түлектерді кездесулерге, конференцияларға немесе Университетте өтетін кеш қонағы ретінде шақыруға болады.

Главная / Модуль панели

Управление панелью

**СООБЩЕНИЕ ДНЯ**  
Статус: Активный  
[Создать новый]

**ЗАМЕТКИ**  
2  
Управление заметками  
[Создать новый]

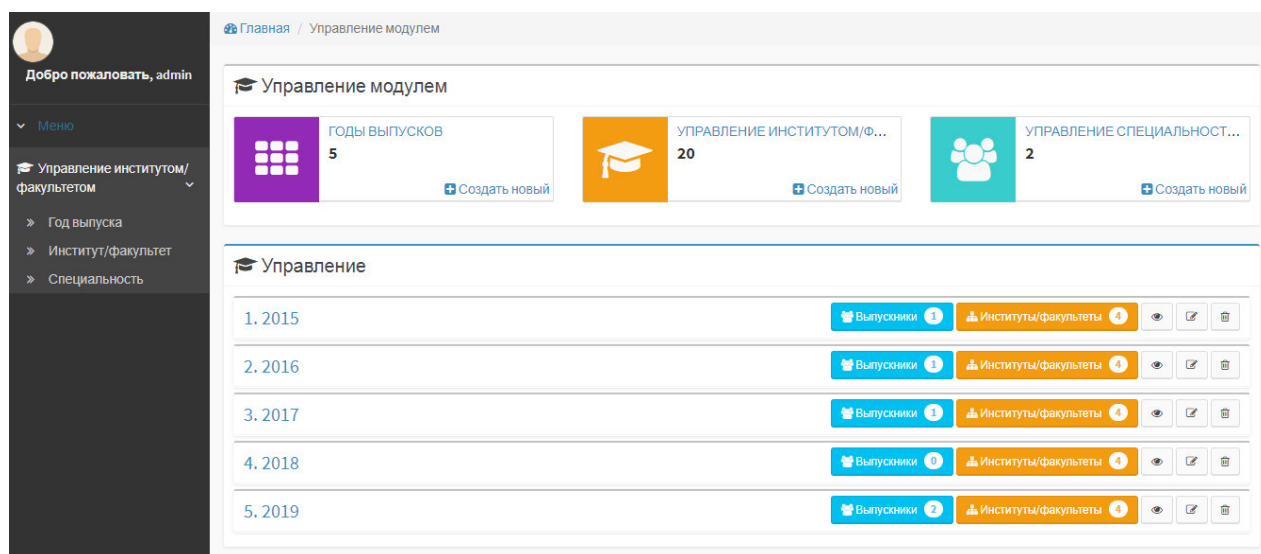
Управление расписанием событий

Сегодня < > май 2019 [Месяц] [Неделя] [День]

Нед	пн	вт	ср	чт	пт	сб	вс
18	29	30	1	2	3	4	5
19	6	7	8	9	10	11	12

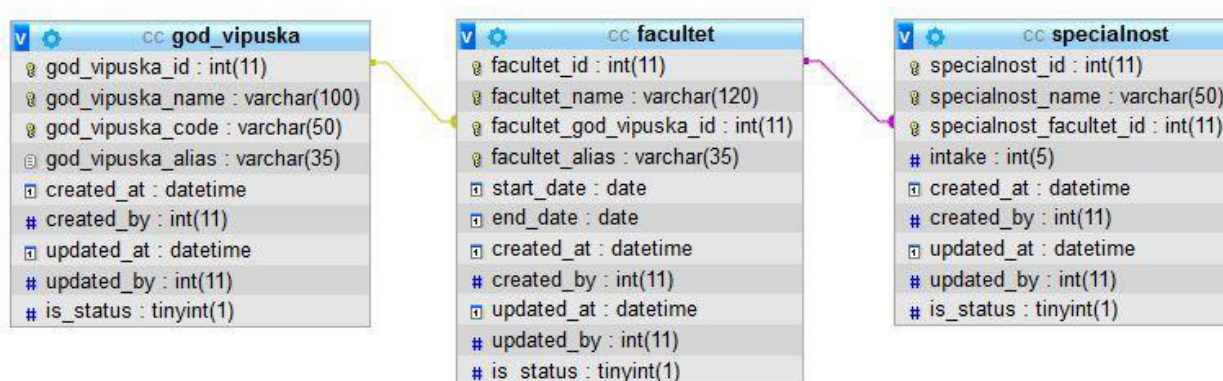
### 3.18-сурет – «Панель» бөлімі

3.19-суретте көрсетілгендей институт/факультет, мамандықты басқару бөлімі арқылы жүйеде қанша түлек тіркелген, соның ішінде қай жылы бітіргендер, қай мамандықты оқыған түлектер бар екенін қадағалап отыруға болады.



3.19-сурет – «Бітіру жылы, институт, мамандықты басқару»

Жүйеде түлектерді тіркеу үшін, олар туралы мәлімет енгізу керек. Яғни, аты-жөні, оқыған мамандығы, факультеті, қазіргі мекен-жайы, жұмыс орны және тағы сол сияқты ақпаратты енгізу қажет (3.21-сурет). Бұл жерде бітіру жылын таңдағанда сол жылғы институт немесе факультет көрсетіледі, ал институт немесе факультетке қарай мамандықтарды таңдауға болады және оны басқа кестемен байланыс жасайтын кестенің сыртқы кілті FOREIGN KEY арқылы жүзеге асырдым (3.20-сурет).



3.20-сурет – Оқу жүйесіне қатысты байланыс диаграммасы

Поля с \* обязательны.

### 1 Личные данные

ID выпускника \*

Заглавие

Имя \*

Отчество

Фамилия \*

Пол

Почта

Номер мобильного

Дата рождения \*

Категория выпуска

Статус

### 2 Информация об образовании

Год выпуска \*

Институт/факультет \*

Специальность \*

Дата присоединения \*

Диплом

Создать

Отмена

### 3.21-сурет – «Түлекті енгізу» процесі

Статистика бөлімі арқылы нақты ақпарат ала аламыз, мысалы, бізге 2019 жылы ЕТЖБҚЕ мамандығын бітірген түлектердің есімдері, байланыс нөмерлері және электронды почтасы қажет. Ол үшін түлектер жайлы ақпаратқа кіріп, бізге керекті пунктерді таңдап «Іздеу» батырмасын басамыз (3.22-сурет). Нәтижесі 3.23–суретте көрсетілген және оны PDF немесе Excel-ге түрлендіріп, сақтауға болады. Дәл сол сияқты менеджерлер туралы ақпарат алуға болады (3.24, 3.25-суреттер).

### 1 Информация о выпускнике

#### Критерии выбора

Год выпуска \*

Институт/факультет

Специальность

Пол

Проверить все

Не выпускника

Фамилия

Отчество

Имя

Пол

Дата рождения

Почта

Группа крови

ИИН

Группа

Дата присоединения

Знания языков

Год выпуска

Институт/факультет

Специальность

Не мобильного

Временной адрес

Постоянный адрес

Искать

### 3.22-сурет – Түлектерді іздеу

Информация о выпускнике					
№	Имя	Отчество	Фамилия	Почта	Номер мобильного
1	Гульмаржан	Талгатовна	Калидоллина	k.inag@mail.ru	87055078417

### 3.23-сурет – Түлектерді іздеу нәтижесі

Информация о менеджере

Критерии выбора

Кафедра \*  
Кафедра IT-инжиниринга

Должность  
--- Выберите должность ---

Категория  
--- Выберите категорию ---

Проверить все

Не менеджера  
 Повышение квалификации, курсы  
 Должность  
 Группа крови  
 Дата присоединения

Фамилия  
 ID карточка  
 Ссылка  
 Пол  
 Не мобильного

Отчество  
 Семейное положение  
 Временной адрес  
 Дата рождения  
 Рабочий телефон

Имя  
 Кафедра  
 Постоянный адрес  
 Не банковского счета  
 Почта

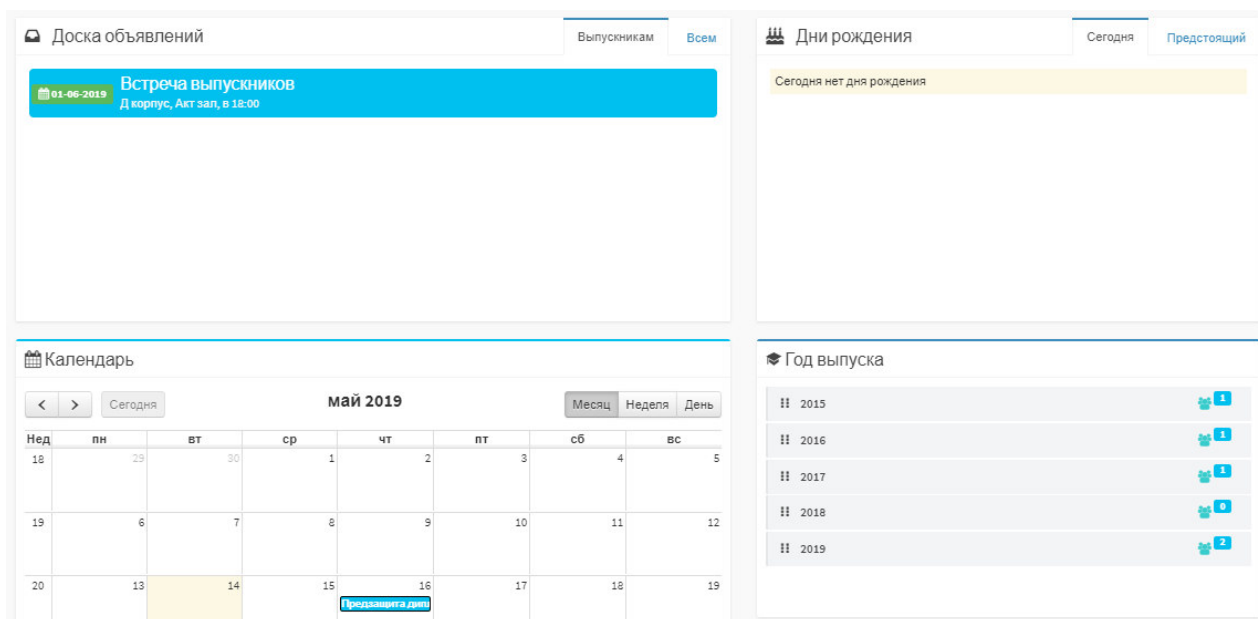
Искать

### 3.24-сурет – Менеджерді іздеу

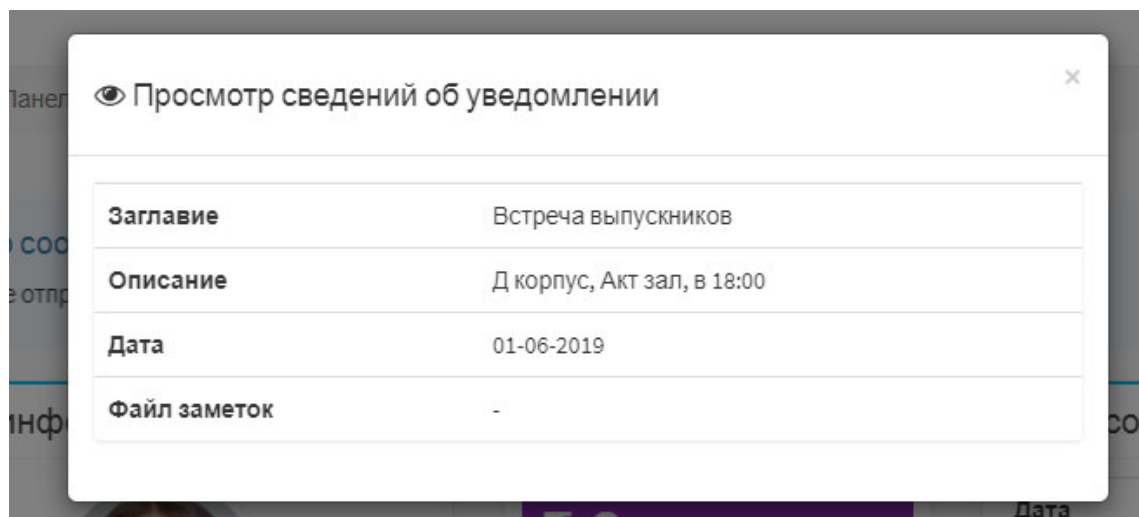
Информация о менеджере												
№	ID менеджера	Имя	Отчество	Фамилия	Повышение квалификации, курсы	ID карточка посещаемости	Диплом	Кафедра	Должность	Ссылка	Временной адрес	Почта
1	1	Салтанат	Тураровна	Сахиева		8596452563	С отличием	Кафедра IT-инжиниринга	Ассистент профессора		3 микрорайон 37а Алматы, Город	XI Акто

### 3.25-сурет – Менеджерді іздеу нәтижесі

Менеджер түлектерге ескертпе ретінде хабарламалар панельге енгізіп отырады (3-26-сурет). Мысалы, «Түлектер кездесуі» болады делік, оның өтетін орны, уақытын жазып, панельге шығарамыз. Түлек өзінің жеке кабинетіне кірген кезде панельден қалқыма терезе арқылы көре алады (3.27-сурет).

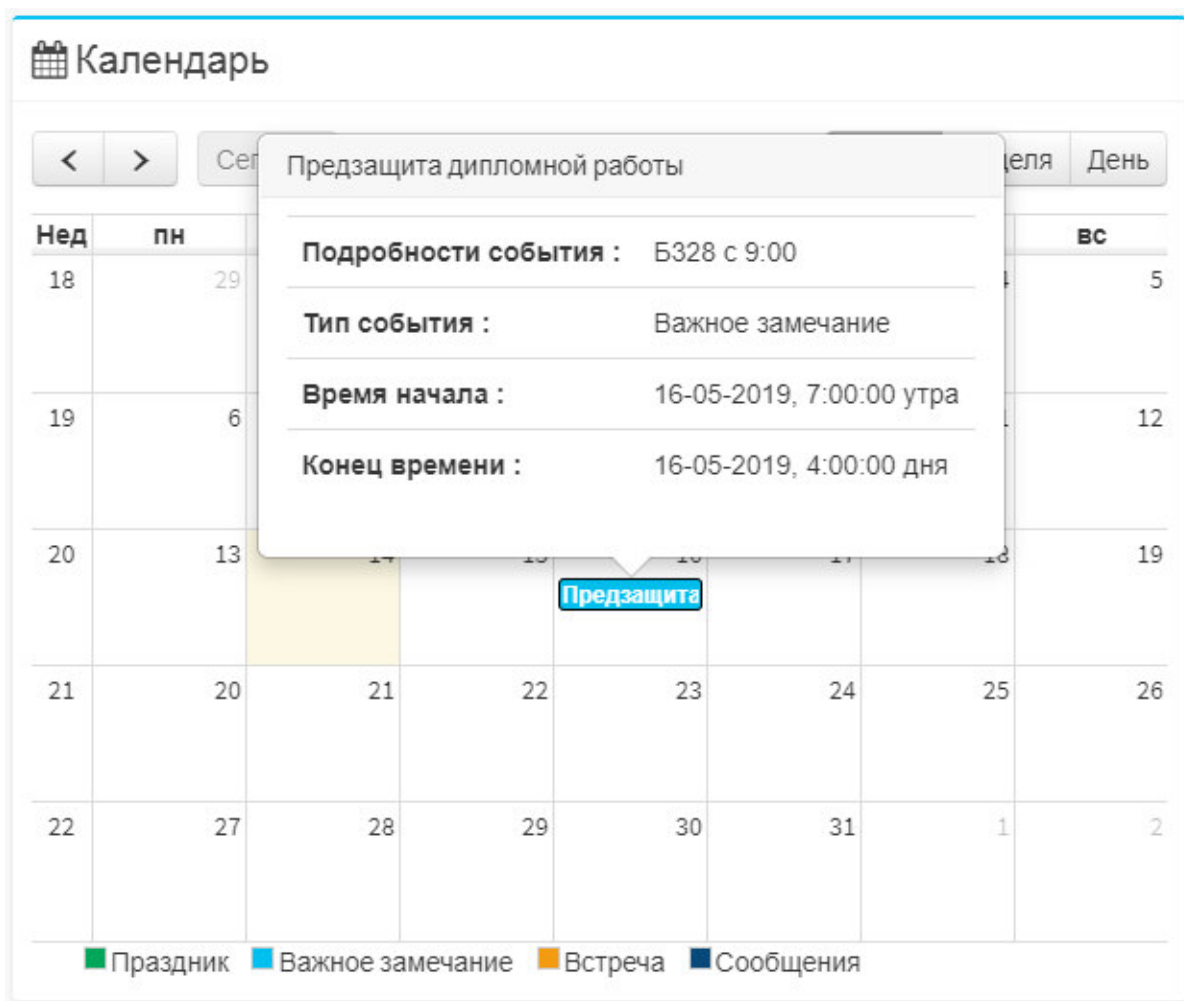


3.26-сурет – Оқиғалар панелі



3.27-сурет – Хабарлама қалқыма терезесі

Белгілі бір маңызды іс-шара, мереке, кездесу және т.б. күнтізбеге енгізіп қою тиімді, себебі жұмыс істейтін адамдардың сол күнге өз кестесін ыңғайластырады. Сондықтан күнтізбеге оқиға енгізу керек (3.28-сурет).



3.28-сурет – Күнтізбедегі оқиға көрінісі

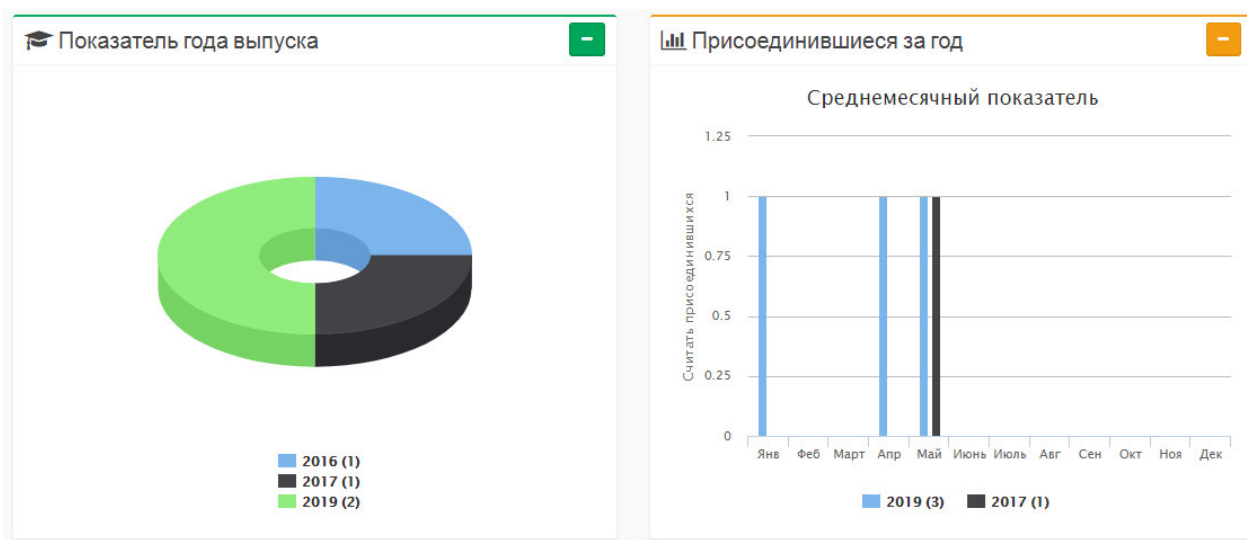
Сайттарда әртүрлі диаграммаларды пайдалануға болады, бұл деректерді пайдаланушыға көрнекі және қол жетімді жеткізуге мүмкіндік береді. Бұрын мұндай графиктер Excel-де жасалған, содан кейін сайтқа сурет салынған. Бірақ біз ақпараттық технологиялар ғасырында өмір сүреміз, сондықтан диаграмманы тез, қарапайым, сонымен қатар интерактивті және әдемі етіп салу керекпіз.

Әрбір графика үшін түстерді, орналасуды, тақырыптарды, анимацияны және т.б. теңшеуге мүмкіндік беретін көптеген параметрлер бар. Егер бұл жұқа болса немесе графиктің стандартты түрін толық қанағаттандырса, аз жинақпен жұмыс істеуге болады.

Мен программаларда да диаграммалар қолдандым, себебі админге менеджер немесе түлектерді бақылау оңайға түседі. Мысалы, диплом көрсеткіштері (3.29-сурет) Университеттегі студенттердің білім сапасын қадағалауға мүмкіндік береді. Ал 3.30-суретте әрбір түлектің бітірген жылы дөңгелек диаграмма түрінде және түлектердің дүйеге кіру көрсеткіші график түрінде көрсетілген.



3.29-сурет – «Диплом көрсеткіші»



3.30-сурет – «Көрсеткіштер»

Оқу орнынан анықтама - нақты ЖОО-да оқу фактісін растайтын құжат. Және тек осы ЖОО осындай құжатты беруге құқылы. Оқу туралы студентке анықтаманы ЖОО тегін беруге міндетті, себебі мұндай анықтаманы беру үшін мемлекеттік баж қарастырылмаған. Анықтама беру оқу орнына байланысты бірнеше аптаға дейін уақыт алуы мүмкін. Ал қашықтықтан оны алу тіпті мүмкін емес, сондықтан веб-бағытталған жүйемде бұл мәселені қарастырдым.

Кез келген түлек бұндай анықтаманы алғысы келсе, сайтқа кіріп өтініш толтырады, өтініш админ почтасына барады және түлектің ЖОО-да оқығандығын расталса түлекке логин, пароль беру арқылы жүйе базасына



енгізеді. Түлек сол логин пароль арқылы жеке кабинетке кіріп (3.30-сурет), «Құжаттар» бөлімінен өзіне керекті мәліметтерді ала алады.

3.29-суретте құжат категорияларын енгізу көрсетілген. Қолданушы типін белгілеп жаңадан құжат категориясын енгізуге болады.

**+ Добавить категорию документа**

Категории документов

Тип пользователя \*

Выпускник  Менеджер  Оба

**Создать**

---

**☰ Список категории документов** **PDF** **EXCEL**

#	Категория	Тип пользователя	
1	Сертификат	Оба	🔍 ✎ ✕
2	Справка с дипломом	Выпускник	🔍 ✎ ✕
3	Справка об обучении	Выпускник	🔍 ✎ ✕

3.29-сурет – «Құжат категориясын енгізу»


Ассоциация выпускников АУЭС

Добро пожаловать, STU5

Главная / Панель выпускника

Окно сообщений в день Каждый день-3

**Моя информация**



ID выпускника: 5

Имя: Гульмаржан Калидоллина

Год выпуска: 2019

Специальность: Вычислительная техника и программное обеспечение (

**₹ Сведения о вложениях**

₹ 0  
Общая сумма вложений

₹ 0  
Активные вложения

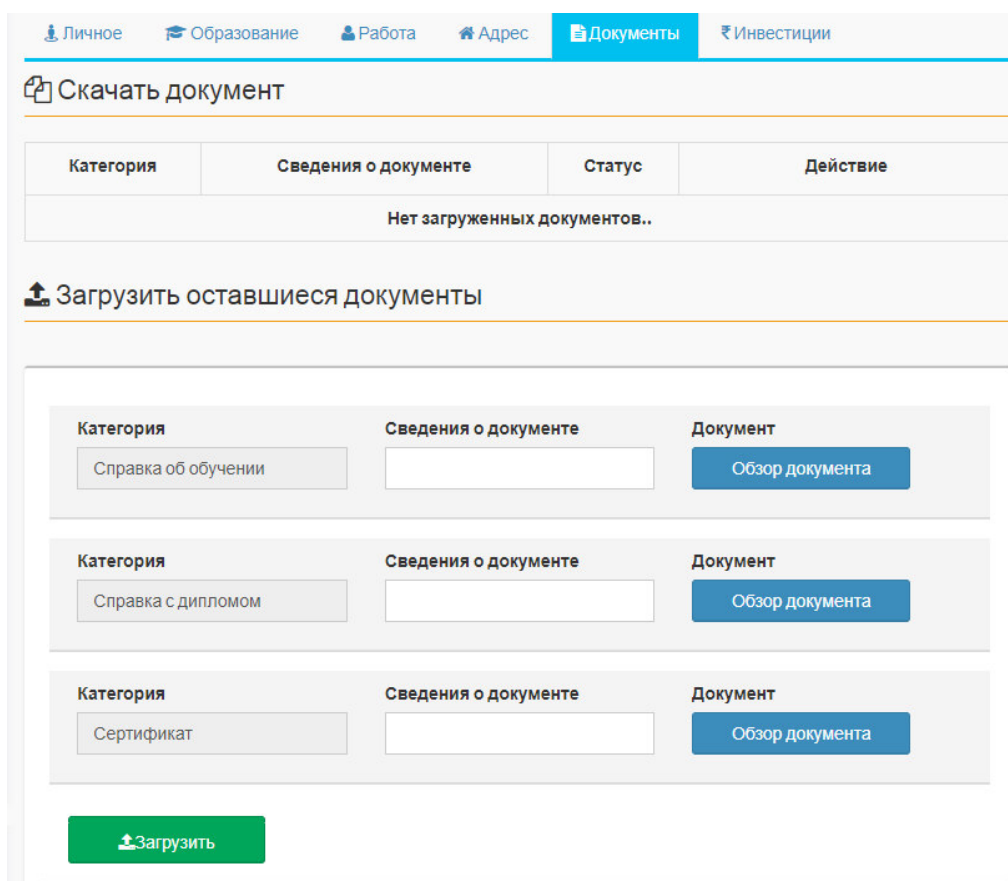
₹ 0

**📅 Список праздников**

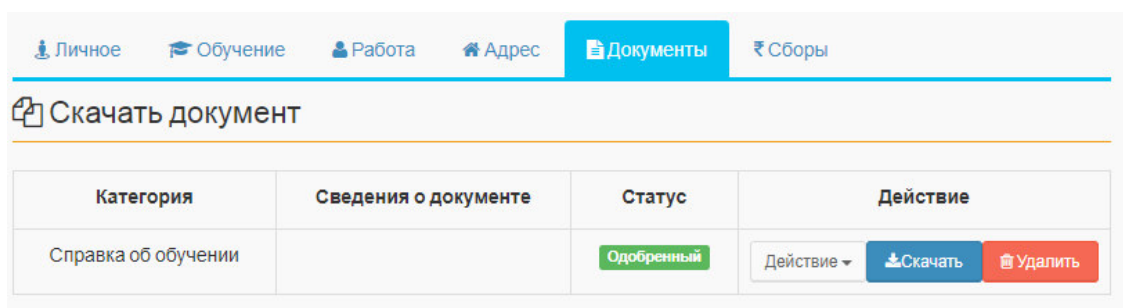
Дата	Праздник
01-01-2019	Новый год
08-03-2019	Международный женский день
22-03-2019	Наурыз мейрамы
01-05-2019	Праздник единства народа Казахстана
07-05-2019	День защитника Отечества
09-05-2019	День Победы
06-07-2019	День Столицы

3.30-сурет – «Түлектің жеке кабинеті»

Админ түлектің профиліне кіріп, құжат енгізуі 3.31-суретте көрсетілген. Құжаттың категориясына қарай «Обзор документа» арқылы құжатты таңдау керек. 3.32-суретте құжаттың енгізілгендегі көрінісі.

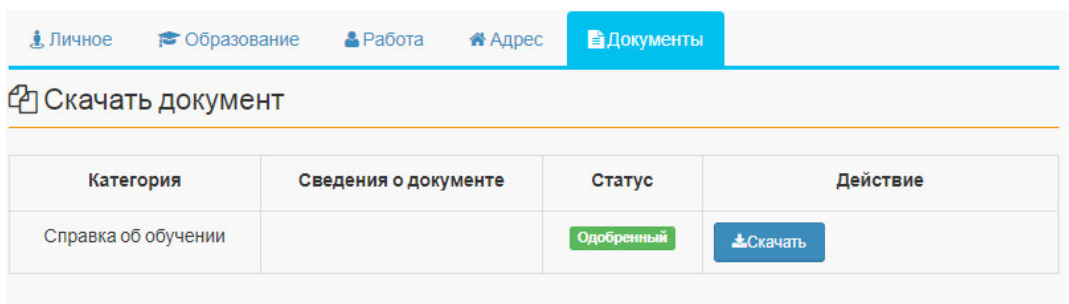


3.31-сурет – Құжатты енгізу процесі

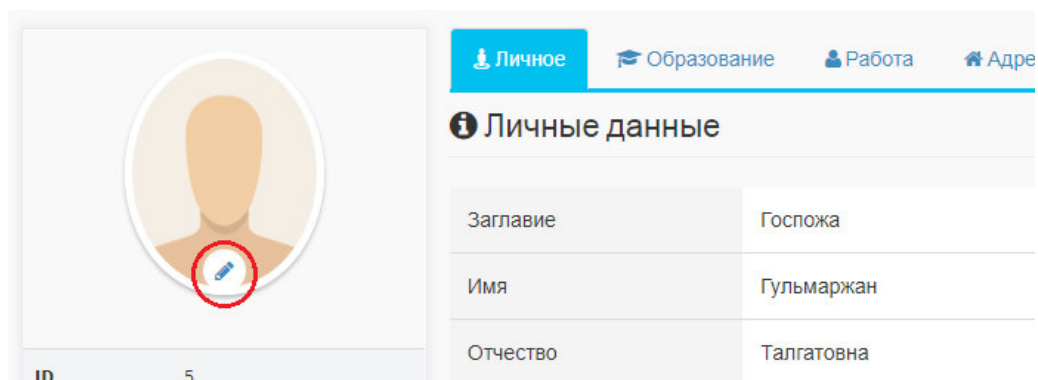


3.32-сурет – Құжат енгізілді

Түлек жеке кабинетіндегі құжаттарды өзіне сақтай алады (3.33-сурет) және өз жеке кабинетінде барлық ақпаратты көре алады, бірақ барлығын бірдей өзгерте алмайды, себебі ондай құқық тек админ, менеджерде ғана. Алайда профильдегі фотоны өзгерте алады, ол үшін 3.34-суретте көрсетілген батырманы басамыз.

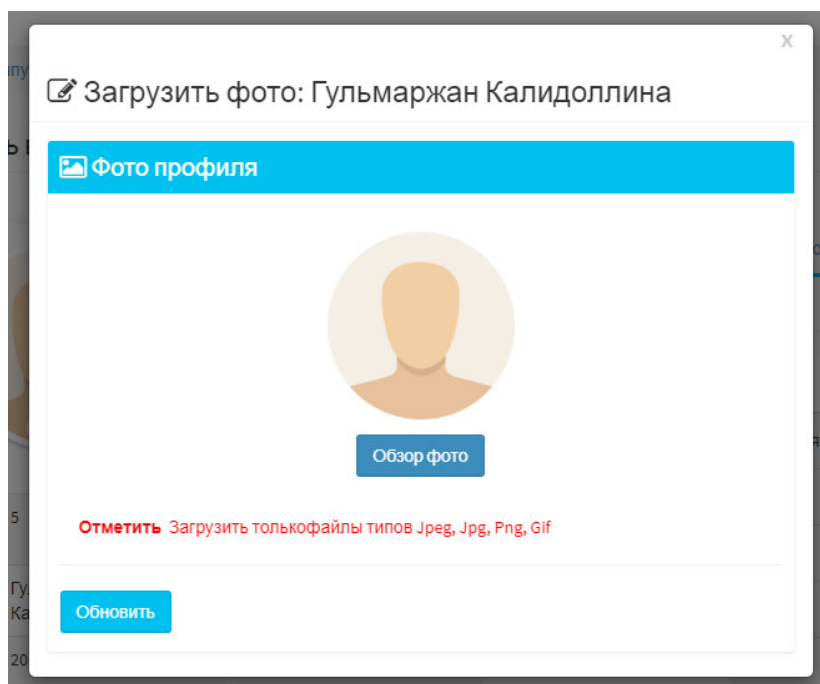


3.33-сурет – Құжатты сақтау мүмкіндігі



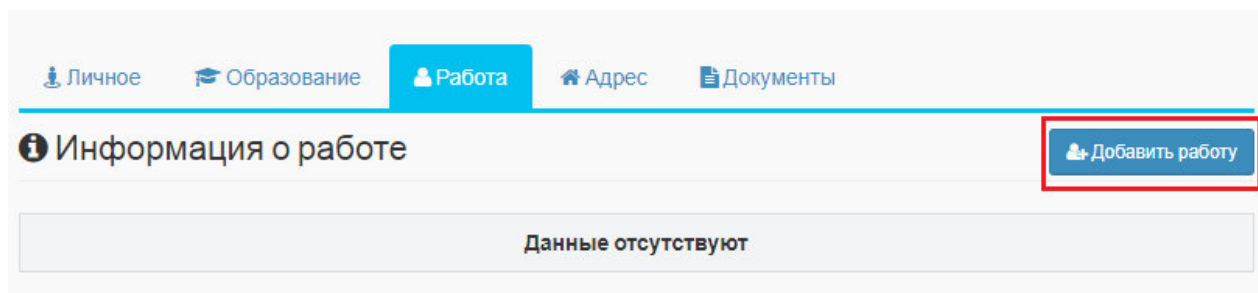
3.34-сурет – Фотоны өзгерту

Батырманы басқан кезде қалқыма терезе шығады және өзіңіздің кез келген суретіңізді қойсаңыз болады (3.35-сурет).



3.35-сурет – Фотоны өзгерту қалқыма терезесі

Түлектердің үнемі қайда жұмыс істейтіндігін админ мен менеджер қадағалап отыруы қиын болғандықтан, ақпаратты түлек өзі басқарып отыра алады (3.36-сурет).



3.36-сурет – Жұмыс туралы ақпаратты басқару

3.37-суретте пароль ауыстыру процесі көрсетілген, яғни парольді жаңартуға түлек құқылы.

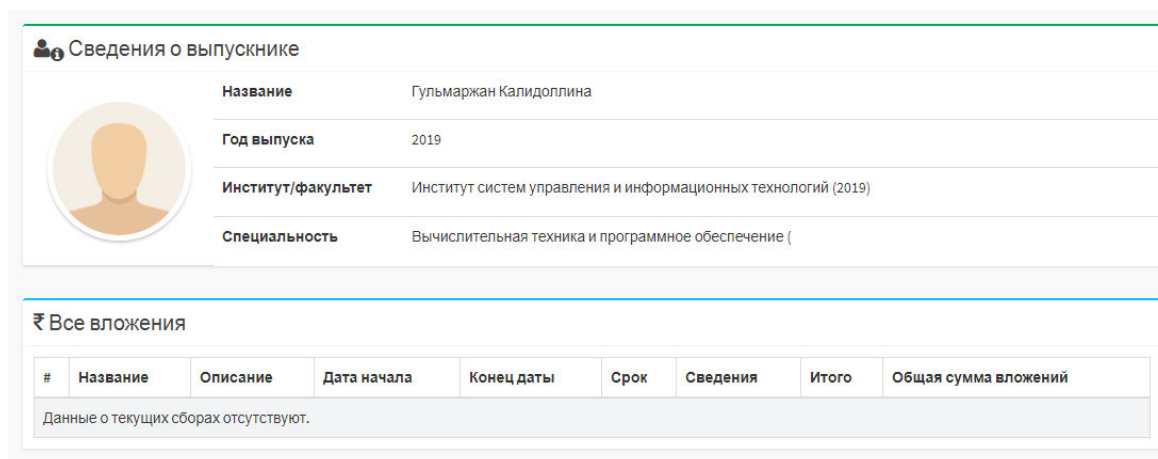
3.36-сурет – Парольді өзгерту

Стартап деп атауға болатын жобалардың ерекшелігі олардың пайда алуға ғана емес, қандай да бір идеяны жақсартуға, қоғам қажеттіліктерін неғұрлым ыңғайлы, оңтайлы шешуді іске асыруға, қызметтерді, өнімдерді және т.б. жақсартуға бағытталуы болып табылады.


Инвесторлар нақты мақсатты көздейді – пайда табу, ал оны әрбір стартап қамтамасыз ете алмайды. Сондықтан инвестициялау туралы шешім есептеу, болжамдар, талдаулар негізінде қабылданады. Стартаптарды мемлекеттік және инвесторлық қолдау жобаның сапалық сипаттамалары, оның табысқа қол жеткізу мүмкіндігі негізінде жүзеге асырылады. Кез келген инвестор, мемлекет немесе бизнес-періште болсын, идеяның болашағы мен мақсаттылығына көз жеткізу керек. Инвесторларды іздеуді бірнеше арналар бойынша жүзеге асыру қажет, өйткені тіпті қаражатты алу сәтсіз әрекеттен жоба бойынша нақты ескертулер түрінде пайда табуға болады. Кез келген стартапер үшін мұндай ақпарат өте құнды болып табылады, өйткені

кемшіліктерді түзете отырып, ол өз жобасының табысқа жетуіне мүмкіндік береді.

Түлектер тарапынан қолдаудың аса маңызды артықшылығы оның сенімділігінің жоғары деңгейі болып табылады. Егер сіздің жобаңыз шын мәнінде пайдалы, тиімді және қоғам тұрғысынан перспективалы екенін, сондай-ақ сіз оны іске асыруға қабілетті және шын мәнінде дайын екенін дәлелдесеңіз, онда сіз оның қаржылық базасы үшін Университетіңіздің түлектері көмектесе алады (3.37-сурет).



**Сведения о выпускнике**

	<b>Название</b>	Гульмаржан Калидоллина
	<b>Год выпуска</b>	2019
	<b>Институт/факультет</b>	Институт систем управления и информационных технологий (2019)
	<b>Специальность</b>	Вычислительная техника и программное обеспечение (

**₹ Все вложения**

#	Название	Описание	Дата начала	Конец даты	Срок	Сведения	Итого	Общая сумма вложений
Данные о текущих сборах отсутствуют.								

3.3-сурет – Түлектің университетке салымы

## 4 Экономикалық бөлім

### 4.1 Экономикалық бөлімде шешілетін мақсаттар мен міндеттер

Экономикалық бөлімнің мақсаты материалдық, еңбек және қаржы шығындарын есептеу болып табылады. Есептеу нәтижесінде қолданбалы программаның өзіндік құны есептелінді.

### 4.2 Қолданбалы программаны әзірлеудің еңбек сыйымдылығын есептеу.

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

ҚБ әзірлеу кезеңдері	Жұмыс түрі	Еңбек сыйымдылығы, адам/сағ.
1 кезең	Жобаға қойылатын талаптарды талдау	20
2 кезең	Техникалық тапсырма	74
3 кезең	Эскиздік жобаны әзірлеу	38
4 кезең	Техникалық жобаны әзірлеу.	96
5 кезең	Программаны әзірлеу	178
6 кезең	Программалық құжаттаманы әзірлеу	136
7 кезең	Өнімді тестілеу	112
8 кезең	Программаны дайындау және тапсыру	44
ҚБ әзірлеудің барлық еңбек сыйымдылығы		698

### 4.3 ҚБ әзірлеуге жұмсалатын шығындарды есептеу

ҚБ әзірлеуге арналған шығындарды анықтау тиісті сметаны жасау жолымен жүргізіледі, ол мынадай баптарды қамтиды:

- а) материалдық шығындар;
- ә) еңбек ақы төлеу шығындары;
- б) әлеуметтік салық;
- в) негізгі қорлардың амортизациясы;
- г) басқа шығындар.

Материалдар және көмекші бөлшектер шығыны ( $Z_M$ ) (4.1) формула бойынша анықталады:

$$Z_M = \sum_{i=1}^n P_i \cdot C_i, \quad (4.1)$$

мұндағы  $P_i$  – материалдық ресурстың  $i$ -түрінің шығыны, заттай бірліктер;

$C_i$  – материалдық ресурстың  $i$ -түрінің бірлігінің бағасы, тг;  
 $i$  – материалдық ресурстың түрі;  
 $n$  – материалдық ресурстар түрлерінің саны.

4.2-кесте – Материалдар және көмекші бөлшектер шығыны

Материалдық ресурстың атауы	Бірлік өлшемі	Саны	Бірлік бағасы, тг	Сомасы, тг
Ноутбук Lenovo 300 series Intel(R) Core(TM) i3-4010U CPU @ 1.70GHz	шт.	1	120000,00	120000,00
БЖ PhpStorm	шт	1	77086,33	77086,33
Барлығы				197086,33

Егер ҚБ әзірлеу үшін электр жабдықтары пайдаланылса, онда 4.3-кестеде келтірілген нысан бойынша электр энергиясына кететін шығындарды есептеу қажет.

Электр энергиясына жұмсалатын шығындардың жалпы сомасы ( $Z_э$ ) (4.2) формула бойынша есептеледі:

$$Z_э = \sum_{i=1}^n M_i \cdot K_i \cdot T_i \cdot C, \quad (4.2)$$

мұндағы  $M_i$  –  $i$ -ші электр жабдығының паспорттық қуаты, кВт;  
 $K_i$  –  $i$ -ші электр жабдығының қуатын пайдалану коэффициенті ( $K_i = 0,7 \div 0,9$ );  
 $T_i$  –  $i$ -ші жабдықтың барлық әзірлеу кезеңіндегі жұмыс уақыты;  
 $C$  – электр энергиясының бағасы, тг / кВт·сағ;  
 $i$  – электр жабдығының түрі;  
 $n$  – электр жабдықтарының саны.

Энергия шығындары программалық қамтамасыз етуді әзірлеу кезеңінің ұзақтығына, программалық жасақтамаға жұмсалатын кВт/сағатқа және 1 кВт/сағ тарифіне негізделген. Заңды тұлғалар үшін Алматы қаласындағы тариф 2019 жылы 1 кВт / сағ үшін 18,32 теңгені құрайды, оның ішінде ҚҚС («АлматыЭнергоСбыт» ЖШС ресми сайтында берілген деректер бойынша).

$$Z_э = 0,7 \cdot 0,7 \cdot 445 \cdot 18,32 = 3994,70 \text{ тг}$$

$$Z_э = 0,3 \cdot 0,7 \cdot 445 \cdot 18,32 = 1712,00 \text{ тг}$$

#### 4.3-кесте – Электр энергиясына кеткен шығын

Жабдық атауы	Паспорттық қуат, кВт	Қуатты пайдалану коэффициенті, кВт	БҚ әзірлеуге арналған жабдықтың жұмыс уақыты, сағ	Электр энергиясы бағасы, тг/кВт×сағ	Сомасы, тг
Ноутбук Lenovo	0,7	0,7	445	18,32	3994,70
Жарықтан дыру	0,3	0,7	445	18,32	1712,00
Барлығы					5706,70

#### 4.4 Еңбек шығындарын есептеу

2019 жылы инженердің орташа жалақысы 210 000,00 тг, ал программист - 120 000,00 теңге (Алматы қаласы үшін).

Қызметкердің айлық жұмыс уақыты (4.3) формула бойынша анықталады:

$$Ч_{\text{м}} = N_{\text{м}} \cdot Ч_{\text{рд}}, \quad (4.3)$$

мұндағы  $Ч_{\text{м}}$  – қызметкердің айлық жұмыс уақыты;  
 $N_{\text{м}}$  – айына жұмыс күндерінің саны;  
 $Ч_{\text{рд}}$  – тәуліктік жұмыс сағаттарының саны.

$$Ч_{\text{м}} = 21 \cdot 8 = 168 \text{ сағ.}$$

Қызметкердің сағаттық ставкасы (4.4) формула бойынша есептелінеді:

$$ЧС_i = \frac{ЗП_i}{ФРВ_i}, \quad (4.4)$$

Инженер-әзірлеуші:

$$ЧС_i = \frac{210000}{168} = 1250,00 \text{ тг}$$

Программист:

$$ЧС_i = \frac{120000}{168} = 714,29 \text{ тг}$$



мұндағы  $ЗП_i$  –  $i$ -ші қызметкердің айлық жалақысы, м;  
 $ФРВ_i$  –  $i$ -ші қызметкердің жұмыс уақытының айлық қоры, сағ.

ҚБ әзірлеудің әрбір қызметкер үшін еңбек сыйымдылығын анықтау үшін 4.1 кестедегі деректер пайдаланылады.

Инженер-әзірлеушінің еңбек сыйымдылығы – 424 адам/сағ. (жобаға қойылатын талаптарды талдау, техникалық тапсырма, эскиздік жобаны әзірлеу, программалық құжаттаманы әзірлеу, өнімді тестілеу, программаны дайындау және тапсыру).

$$T_1 = 20 + 74 + 38 + 136 + 112 + 44 = 424 \text{ адам/сағ.}$$

ҚБ программисттің еңбек сыйымдылығы – 424 адам/сағ. (эскиздік жобаны әзірлеу, техникалық жобаны әзірлеу, программаны әзірлеу, өнімді тестілеу).

$$T_1 = 38 + 96 + 178 + 112 = 424 \text{ адам/сағ.}$$

Еңбек шығындарының жалпы көлемі ( $З_{тр}$ ) (4.5) формула бойынша анықталады:

$$З_{тр} = \sum_{i=1}^n ЧС_i \times T_i, \quad (4.5)$$

мұндағы  $ЧС_i$  –  $i$ -ші қызметкердің сағаттық ставкасы, тг;  
 $T_i$  – ҚБ әзірлеудің еңбек сыйымдылығы, адам/сағ.;  
 $i$  – қызметкер санаты;  
 $n$  – ҚБ-ны әзірлеумен айналысатын қызметкерлер саны.

Инженер-әзірлеуші:

$$З_{тр} = 1250,00 \cdot 424 = 530000,00 \text{ тг}$$

Программист:

$$З_{тр} = 714,29 \cdot 424 = 302858,96 \text{ тг}$$

Барлығы:

$$З_{тр} = 530000,00 + 302858,96 = 832858,96 \text{ тг}$$

Қосымша еңбекақы (4.6) формула бойынша анықталады:

$$З_{доп} = З_{тр} \cdot 10\%, \quad (4.6)$$

1.4 кесте - еңбекке ақы төлеу құны

Квалификация	ҚБ әзірлеудің еңбек сыйымдылығы, адам/сағ.;	Сағаттық ставка, тг/сағ.	Сомасы, тг
Инженер-әзірлеуші	424	1250,00	530000,00
Программист	424	714,29	302858,96
Барлығы			832858,96

$$З_{\text{доп}} = 832858,96 \cdot 0,1 = 83285,90 \text{ тг}$$

Жалақы қоры (4.7) формула бойынша анықталады:

$$\Phi_{\text{зп}} = З_{\text{тр.о}} + З_{\text{доп}}, \quad (4.7)$$

$$\Phi_{\text{зп}} = 832858,96 + 83285,90 = 916144,86 \text{ тг}$$

Әлеуметтік салық (4.8) формула бойынша анықталады:

$$Н_c = (\Phi_{\text{зп}} - \text{ОПВ}) \cdot 11\%, \quad (4.8)$$

мұндағы ОПВ - міндетті зейнетақы салымы, ол  $\Phi_{\text{зп}}$ -ның 10% -ын құрайды.

#### 4.5 Негізгі қорлардың амортизациясын есептеу

Амортизацияның жалпы сомасы (4.9) формула бойынша анықталады:

$$З_{\text{АМ}} = \sum_{i=1}^n \frac{\Phi_i \cdot Н_{\text{А}i} \cdot Т_{\text{НИР}i}}{100 \cdot Т_{\text{ЭФ}i}}, \quad (4.9)$$

мұндағы  $\Phi$  –  $i$ -ші ОФ-тың құны, тг;  
 $Н_{\text{А}i}$  –  $i$ -ші ОФ-тың жыл сайынғы амортизациясы, %;  
 $Т_{\text{НИР}i}$  – ҚБ-ны әзірлеудің барлық кезеңіндегі  $i$ -ші ОФ жұмысының уақыты, сағ.;  
 $Т_{\text{ЭФ}i}$  – бір жылғы  $i$ -ші ОФ жұмысының уақытының тиімді қоры, жыл / жыл;  
 $i$  – ОФ түрі;  
 $n$  – ОФ саны.

Жылдық амортизацияны (4.10) формула бойынша есептейміз:

$$H_{Ai} = \frac{100}{T_{Ni}}, \quad (4.10)$$

мұндағы  $T_{Ni}$  –  $i$ -ші ОФ-ті пайдаланудың ықтимал мерзімі;

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

ҚБ-ны әзірлеуге арналған программалық қамтамасыз ету уақытын анықтау үшін 4.1-кестеден алынған мәліметтер қолданылады.

ҚБ-ны әзірлеуге арналған PhpStorm программалық жасақтамасының жұмыс уақыты 386 сағатты құрайды (техникалық жобаны әзірлеу, программаларды әзірлеу, өнімді тестілеу).

$$T_1 = 96 + 178 + 112 = 386 \text{ сағ.}$$

Жабдық:

$$З_{AM} = \frac{120000,00 \cdot 25 \cdot 698}{100 \cdot 1920} = 10906,25 \text{ тг}$$

Программалық жасақтама:

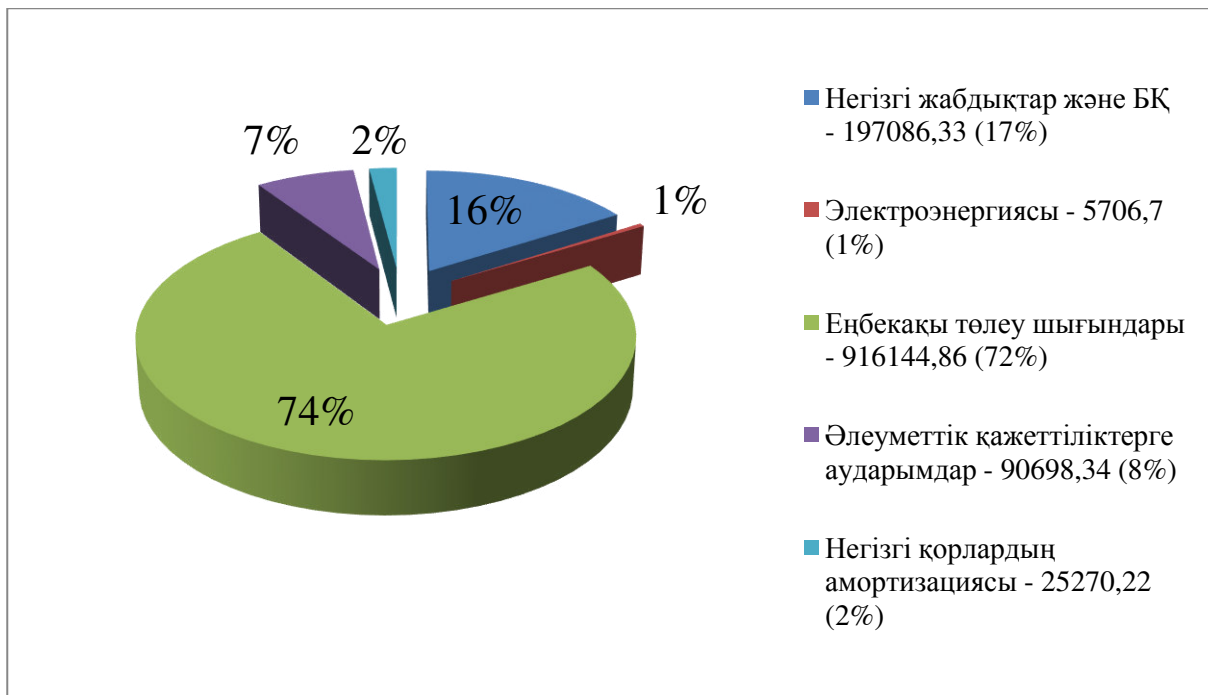
$$З_{AM} = \frac{77086,33 \cdot 25 \cdot 386}{100 \cdot 1920} = 3874,69 \text{ тг}$$

#### 4.5-кесте – негізгі қорлардың амортизациясы

Жабдық пен программалық жасақтама атауы	Бағасы, тг	Жылдық амортизация, %	Жабдықтың және ПҚ тиімді жұмыс уақыты, сағ./жыл	Кеткен уақыты, сағ.	Сомасы, тг
Ноутбук Lenovo	120000,00	25	1920	698	21395,83
БЖ PhpStorm	77086,33	25	1920	386	3874,93
Барлығы					25270,22

#### 4.6-кесте – ҚБ әзірлеуге арналған шығындар сметасы

Шығындар	Сомасы, тг
1. Негізгі жабдықтар және БҚ	197086,33
2. Электроэнергиясы	5706,70
3. Еңбекақы төлеу шығындары	916144,86
4. Әлеуметтік қажеттіліктерге аударымдар	90698,34
5. Негізгі қорлардың амортизациясы	25 270,22
Барлығы	1 151620,55



4.1-сурет – ҚБ әзірлеуге арналған шығындар сметасы

#### 4.6 ҚБ ықтимал (шарттық) бағасын анықтау

Қолданбалы БҚ үшін шарттық баға (4.11) формула бойынша есептеледі:

$$Ц_d = Z_{\text{НИР}} \cdot \left(1 + \frac{P}{100}\right), \quad (4.11)$$

мұндағы  $Z_{\text{НИР}}$  - ПҚ әзірлеуге арналған шығындар (4.6-кесте), тг;  
 $P$  - БҚ рентабельділігінің орташа деңгейі – 25%.

$$Ц_d = 1151620,55 \cdot (1 + 0,25) = 1439525,69 \text{ тг}$$

Одан кейін, сату бағасы қосымша құн салығын (ҚҚС) ескере отырып анықталады, ҚҚС ставкасы Қазақстан Республикасының Салық кодексінде белгіленеді. 2019 жылы ҚҚС ставкасы 12% деңгейінде белгіленеді.

ҚҚС қоса алғанда, сату бағасы (4.12) формула бойынша есептеледі:

$$Ц_p = Ц_d + Ц_d \cdot \text{ҚҚС}, \quad (4.12)$$

$$Ц_p = 1439525,69 + 1439525,69 \cdot 0,12 = 1612268,77 \text{ тг}$$

Пайданы (4.13) формула бойынша өзіндік құнның 25% мөлшерінде аламыз:

$$П = Z_{\text{НИР}} \cdot 0,25, \quad (4.13)$$

$$П = 1151620,55 \cdot 0,25 = 287905,14 \text{ тг}$$

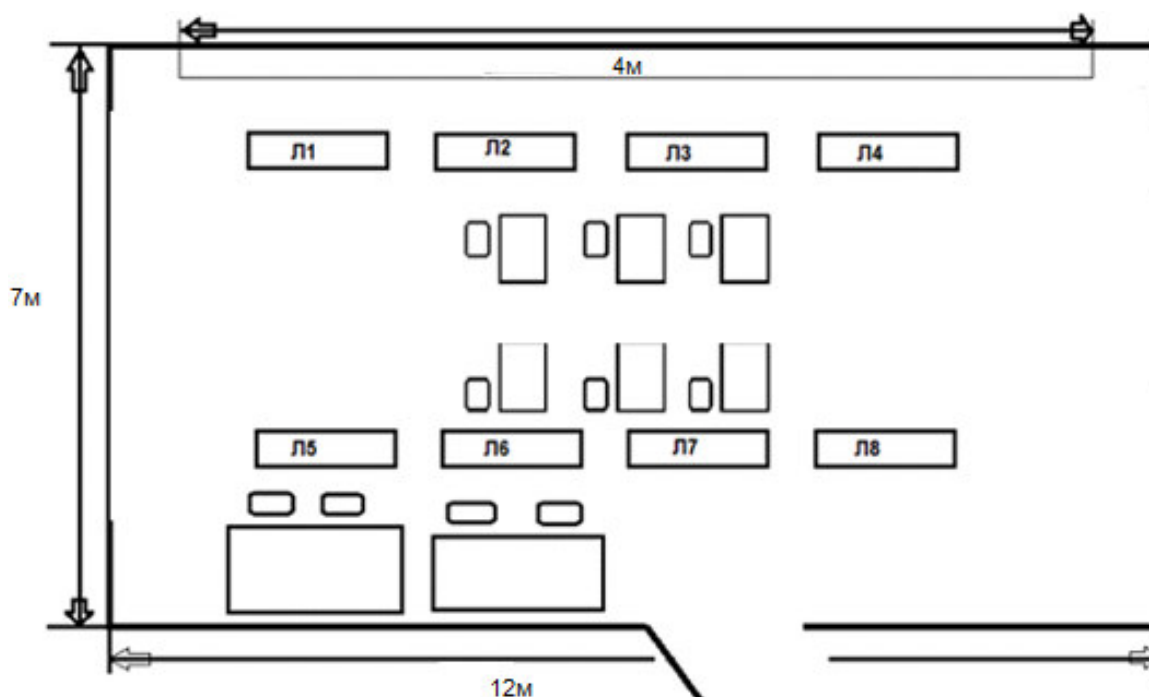
#### **4.7 Экономикалық бөлімі бойынша қорытынды**

Бұл бөлімде экономикалық тиімділіктің есебі жүргізілді, ол жоғарыда көрсетілген программа әзірлеуге кететін уақыт және шығындар есептелінді. Сатып алу бағасы ҚҚС есебімен 1612268,77 теңге, өзіндік құны 1151620,55 теңге, алынатын пайда 287905,14 теңгеге тең. Шығынның негізгі бөлігін еңбекақы төлеу шығындары құрайды (74%).

Жүргізілген есептеулердің нәтижелері Университет түлектерінің қауымдастығы үшін веб-бағытталған жүйе рентабельді және экономикалық негізделген болып табылатынын, ішкі нарықта бәсекеге қабілеттілігін қамтамасыз ететінін көрсетеді.

## 5 Өміртіршілік қауіпсіздігі бөлімі

Программисттің жұмыс орны Университеттің 4-қабатында орналасқан және 1-суретте көрсетілген. Жұмыс бетінің еден деңгейінен биіктігі 1 м, бөлмеде 2 терезе бар және 1 м биіктіктен басталады, терезенің ені 3,5 м және биіктігі 2 м, жарық ойықтарының ауданы әр терезе үшін  $S=7 \text{ м}^2$ , көру жұмыстарының разряды – III. Жасанды жарықтандыру қуаты 65 Вт, жарық ағыны 3750 лм болатын ЛД-65-7 G13 разрядты төмен қысымды люминесцентті шамдармен ұсынылған. 8 жұмыс орны және сәйкесінше сонша жабдықтар орналасқан. Бөлмедегі ауаның қажетті мөлшері нормаларға сәйкес келеді. Бөлмеде шулы аспаптар жоқ.



5.1-сурет – Программисттің жұмыс орны

### 5.1 Табиғи жарықтандыруды есептеу

Бірінші кезекте жарық ойықтарының ауданы 2 терезе үшін  $S=7 \text{ м}^2$  бөлменің қалыпты жарықтандырылуы үшін жеткілікті ме соны тексереміз. Есептеу (5.1) формула бойынша бүйірлік жарықтандыру кезінде жарық ойықтарының ауданын алдын ала анықтаудан тұрады:

$$100 \frac{S_0}{S_n} = \frac{e_N K_3 \eta_0}{\tau_0 r_1} K_{зд}, \quad (5.1)$$

мұндағы  $S_0$  – бүйірлік жарықтандыру кезіндегі жарық ойықтарының ауданы,  $\text{м}^2$ ;

$S_n$  – еденнің ауданы,  $\text{м}^2$ ;

$n$  – ТЖК нормаланған мәні;  
 $K_3 = 1,2$  – жарық өткізетін материалдың тік орналасуы кезіндегі қор коэффициенті;  
 $\tau_0$  – жарық өткізудің жалпы коэффициенті;  
 $r_1$  – бөлме бетінен шағылысқан жарықтың және ғимаратқа жанасатын төсеніш қабаттың арқасында бүйірлік жарықтандыру кезінде ТЖК көтерілуін ескеретін коэффициент;  
 $K_{зд} = 1$  – қарсы тұрған ғимараттардың терезелерді қараңғылауын ескеретін коэффициент.

Еден ауданы (5.2) формула бойынша анықталады:

$$S_n = L \cdot B, \quad (5.2)$$

Еден ауданы (5.2) формула бойынша тең:

$$S_n = 12 \cdot 7 = 84 \text{ м}^2$$

Табиғи жарықтандыру коэффициентінің нормаланған мәні (5.3) формула бойынша есептеледі:

$$e_N = e_n \cdot m_N, \quad (5.3)$$

мұндағы  $N$  – табиғи жарықпен қамтамасыз ету жөніндегі әкімшілік-аумақтық аудан тобының нөмірі;

$e_n = 1,2$  – табиғи жарықтандыру коэффициентінің мәні;

$m_N = 0,65$  – жарықтық климаттың коэффициенті.

(5.3) формуласына сәйкес әртүрлі аумақтарда орналасқан ғимараттар үшін  $e_N$  нормаланған мәні тең:

$$e_N = e_n \cdot m_N = 1,2 \cdot 0,65 = 0,78 \%$$

Бір жақты жарықтандырылған бөлменің тереңдігі:

$$l = B - 1 = 7 - 1 = 6 \text{ м}$$

$\eta_0$  жарық сипаттамасының мәнін алу үшін келесі қатынастар есептелді:

$$\frac{L}{l} = \frac{12}{6} = 2$$

$$\frac{1}{h_{\text{расч}}} = \frac{1}{(h_{\text{ок}} + h_{\text{нок}}) - h_{\text{пп}}} = \frac{6}{(2 + 1) - 1} = 3$$

Жарық сипаттамасының мәні:

$$\eta_0 = 9,5 + \frac{(10,5 - 9,5) \cdot (3 - 1)}{3 - 2} = 11,5$$

Жарық өткізудің жалпы коэффициенті ( $\tau_0$ ) (5.4) формуламен анықталады:

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 \quad (5.4)$$

мұндағы  $\tau_1 = 0,8$  – материалды жарық өткізу коэффициенті, терезе әйнегі үшін;

$\tau_2 = 0,75$  – терезенің байланыстарында жарықтың жоғалуын ескеретін коэффициент;

$\tau_3 = 1$  – бұл тірек конструкцияларындағы жарықтың жоғалуы есепке алынған, бүйірлік жарықтандыру кезіндегі коэффициент;

$\tau_4 = 1$  – күн сәулесінен қорғайтын құрылғылардағы жарықтың жоғалуын ескеру коэффициенті.

(5.4) формулаға мәндерді қойып, ( $\tau_0$ ) табамыз:

$$\tau_0 = 0,8 \cdot 0,75 \cdot 1 \cdot 1 = 0,6$$

Бүйірлік жарықтандыру коэффициентін ( $\tau_1$ ) табамыз. Бұл үшін бөлменің тереңдігін әдеттегі жұмыс бетінің деңгейінен терезенің жоғарғы жағына дейін қатынасын анықтаймыз:

$$\frac{1}{h_{\text{расч}}} = \frac{6}{3} = 2$$

Есептік нүкте мен сыртқы қабырға арасындағы бөлменің тереңдігіне ара қатынасы:

$$\frac{1}{B} = \frac{6}{7} = 0,86$$

Бөлменің ұзындығына тереңдікке қатынасы:

$$\frac{L}{1} = \frac{12}{6} = 2$$

Төбенің, қабырғалардың және еденнің орташа өлшенген коэффициенті (5.5) формула бойынша анықталады:



$$\rho = \frac{\rho_{\text{пот}} + \rho_{\text{ст}} + \rho_{\text{пол}}}{3} \quad (5.5)$$

Осылайша (5.5) формула бойынша:

$$\rho = \frac{50+30+10}{3} = 30\% = 0,3$$

Алынған мәндерден бүйірлік жарықтандыру коэффициентін ( $r_1$ ) табамыз.

$$r_1 = 1,7 + \frac{(2-1,7) \cdot (2-1)}{2-1} = 2$$

Барлық формулалардан табылған мәндерді ала отырып (5.1) формула бойынша біз бүйірлік жарықтандыру кезіндегі  $S_0$  жарық саңылауларының ауданын табамыз:

$$S_0 = \frac{84 \cdot 0,78 \cdot 1,2 \cdot 11,5 \cdot 1}{100 \cdot 0,6 \cdot 2} \approx 7,5 \text{ м}^2$$

Бұл бөлмеде жарық ашу аймағы тексерілді, нақты терезе жарықтары жақсы жарықтандыратына көзіміз жетті. Осы деректерден қосымша жарық көздері қысқы және түнде керек болатынын анықтадық.

## 5.2 Жасанды жарықтандыруды тексеру

Бөлмеде жасанды жарықтандыру қуаты 65 Вт жарық ағыны 3570 лм люминесцентті лампалар мен диаметрі 38 мм және ұзындығы 1515 мм шырағандар арқылы жүзеге асырылған. Жарықтандыру үшін ЛД65-7 G13 түріндегі өлшемі 1540x240x170 мм шамдар пайдаланылады. 300 лк жарықтандыруды қажет ететін қолдану коэффициентін пайдаланып шамдардың санын есептеп шығарамыз, бұл III-ші санаттағы визуалды жұмыстарды қамтамасыз ету үшін жеткілікті.

Шамдардың санын есептеу үшін (5.6) формула пайдаланылады:

$$N_{\text{л}} = \frac{S_{\text{помещ}} \cdot K_z \cdot Z \cdot E_{\text{н}}}{\Phi_{\text{л}} \cdot n \cdot \eta}, \quad (5.6)$$

мұндағы  $Z$  – ең аз жарықтану коэффициенті (орташа және ең аз жарықтану қатынасы), есептерде  $z$  коэффициенті 1,1÷1,2 шегінде қабылданады;

$n = 2$  – шырағандағы шам саны;

$\eta$  – бөлменің индексіне және шағылысу коэффициенттеріне байланысты пайдалану коэффициенті;

$K_3 = 1,3$  – шаң бөлінбейтін үшін жерлер қор коэффициенті.

Жарықтандыру қондырғысын пайдалану кезінде жұмыс беттеріндегі жарықтандыру жарық көздерінің жарық ағынын азайту, шамдар мен жарықтандыру шырағданы ластануы, сондай-ақ жарықтандырылатын бөлме қабырғалары мен төбесінің ластануы есебінен төмендейді. Сондықтан жарықтандыру қондырғысының қуатын анықтау кезінде қор коэффициенті енгізіледі. Қор коэффициенті ауаның шаңмен, түтінмен, найзамен және т. б. ластану дәрежесіне байланысты.

Жарықтандырудың бірқалыпты еместігін сипаттайтын  $Z$  коэффициенті көптеген айнымалы функциялар болып табылады және ең үлкен дәрежеде шамдардың арасындағы қашықтықтың шамдардың ілінуінің есептік биіктігіне ( $L/h$ ) қатынасына байланысты болады, оны ұлғайтумен ұсынылған  $Z$  мәнінен жоғары күрт өседі.  $\lambda$  ұсынылған мәндерден аспайтын жағдайда қабылдауға болады:

$Z = 1,15$  – қыздыру шамдары үшін;

$Z = 1,1$  – люминесцентті шамдар үшін.

Жарықтану қондырғысын пайдалану коэффициенті  $\eta$  – бұл жұмыс бетіне түсетін жарық ағынының, жіберілетін жарық ағынына қатынасы.

Қолдану коэффициенті шамдардың түріне, бөлменің геометриялық өлшемдеріне және бетінің көрсететін коэффициенттеріне байланысты болады.

Пайдалану коэффициентін анықтау үшін, бөлме индексін (5.7) формула арқылы анықтау керек:

$$i = \frac{S_{\Pi}}{(L+B)h_{\text{расч}}} = \frac{A \cdot B}{(L+B)h_{\text{расч}}}, \quad (5.7)$$

мұндағы  $S$ ,  $L$ ,  $B$  тиісінше ауданы, ұзындығы және ені.

(5.7) формулаға сәйкес мәндерді қоямыз:

$$i = \frac{12 \cdot 7}{(7+5) \cdot 2} = 3,5$$

Бір шамның қажетті жарық ағыны (5.8) формула бойынша анықталады:

$$\Phi_{\text{л}} = \frac{E_{\text{min}} \cdot S \cdot Z \cdot k}{N \cdot \eta \cdot n}, \quad (5.8)$$

мұндағы  $E_{\text{min}}$  – ең аз нормаланған жарықтандыру, лк;

$k$  – қор коэффициенті (люминесцентті шамдар үшін  $k=1,3$ );

$S$  – жарықтандырылатын алаң, м<sup>2</sup>;

$Z$  – ең аз жарықтандыру коэффициенті (жарықтандырудың біркелкі емес коэффициенті);

$N$  – шамдардың саны;

n – Шамдағы шамдардың саны;  
h – бірлік үлесіндегі жарық ағынын пайдалану коэффициенті.

потолок	80	80	80	70	50	50	30	0
стены	80	50	50	50	50	30	30	0
пол	30	30	10	20	10	10	10	0
0.6	65	43	34	41	40	34	33	28
0.8	74	53	43	50	48	42	41	36
1	81	60	49	57	54	48	48	42
1.25	87	69	57	64	61	56	55	49
1.5	91	74	62	69	65	60	59	54
2	96	82	68	76	70	66	65	60
2.5	100	87	73	80	74	71	70	65
3	102	92	77	84	78	75	73	69
4	105	96	80	87	80	78	76	72
5	106	99	83	90	82	80	79	75

5.2-сурет – Люминесцентті шамдардың жарық ағынын пайдалану коэффициенті

Шырағдан тобы 1-ге тең болғандықтан 5.2-суреттен  $\eta$  табамыз.  $\eta = 0,42$ .  
Жасанды жарықтандыру жеткілікті ме тексеру үшін  $E_{min}$  есептеу керек, ол 300 лк көп болуы керек.

$$E_{min} = \frac{3570 \cdot 8 \cdot 2 \cdot 0.42}{84 \cdot 1.1 \cdot 1.3} = 200 \text{ лк}$$

$E_{min}$  300лк-ден аз. Бұл мәселені шешу керек. Мен жарық ағыны 3570 лм қуаты 65Вт бойынша ескі люменцентті шамдарды 4250 лм жарық ағыны бар қуаты 80Вт жаңа люменцентті шамдарға ауыстыруды шештім. Содан кейін  $E_{min}$  тағы да тексердім.

$$E_{min} = \frac{4250 \cdot 8 \cdot 2 \cdot 0.42}{84 \cdot 1.1 \cdot 1.3} = 237 \text{ лк}$$

Бұнымен мәселе шешілмеді. Сондықтан шырағдандағы шамдардың санын көбейтуді шештім, енді әрбір шырағданда 3 шамнан:

$$E_{min} = \frac{4250 \cdot 8 \cdot 3 \cdot 0.42}{84 \cdot 1.1 \cdot 1.3} = 356 \text{ лк}$$

Нәтижесінде біз бұл бөлмені жақсы жарықтандыру үшін жеткілікті болды. Шамдардың қуатын арттыру керек, яғни жалпы жарықтандыру шамдарында 4250 лм жарық ағыны бар қуаты 80 Вт шамды таңдау және шырағдандағы шамдардың санын арттыру керек.

Жұмыс орнының үстінде шамдарды ілу биіктігін (5.9) формула бойынша анықтаймыз:

$$h_{\text{расч}} = H_{\text{пом}} - (H_{\text{св}} + H_{\text{р.п.}}) , \quad (5.9)$$

мұндағы  $H_{\text{св}} = 0,2$  – шамның іліну биіктігі, м;  
 $H_{\text{р.п.}} = 1$  – еден үстін мен жұмыс бетінің қашықтығы, м;  
 $H_{\text{пом}} = 4$  – бөлме биіктігі, м.

Шырағданның іліну ұзындығы (5.9) формула бойынша:

$$h_{\text{расч}} = 4 - (0,2 + 1) = 2,8 \text{ м}$$

Шырағдандардың арасындағы қажетті қашықтықты (5.10) формула бойынша анықтаймыз:

$$L = \lambda \cdot h , \quad (5.10)$$

мұндағы  $L$  – көрші шамдардың немесе люминесцентті шамдардың қатарлары арасындағы қашықтық;  
 $h$  – шамды жұмыс бетінен ілу биіктігі.

Люминесцентті шамдары бар шырағданды қолданған кезде  $\lambda = 1.2 \div 2.4$ .

Шырағдандардың арасындағы қажетті қашықтық:

$$L_A = \lambda \cdot h = 1,07 \cdot 2,8 = 3 \text{ м}$$

Шырағдандардың қатарларының арасындағы қашықтық:

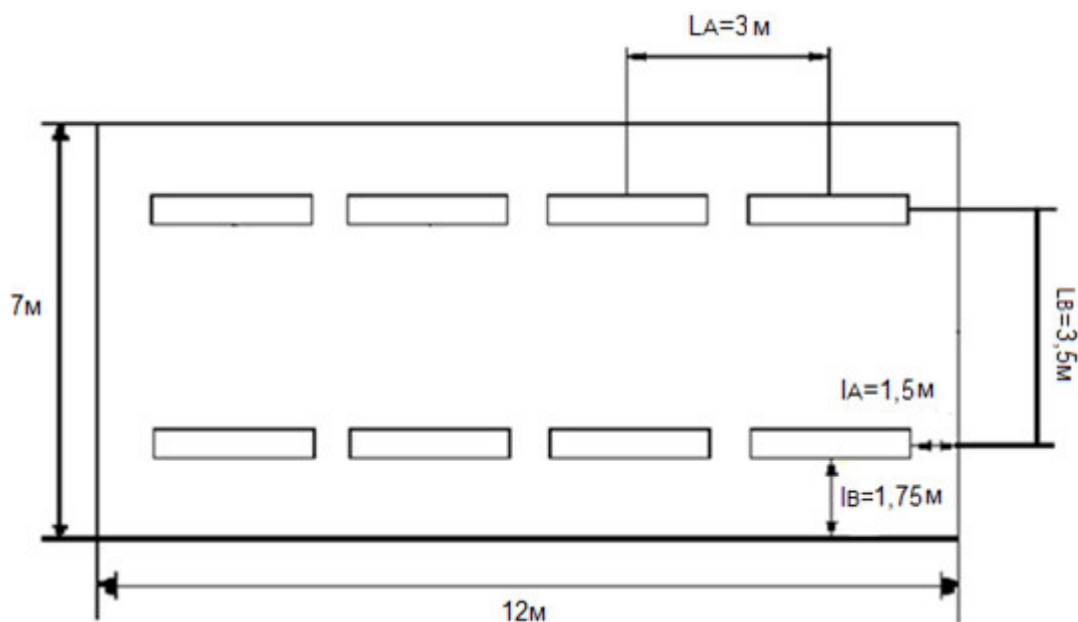
$$L_B = \lambda \cdot h_p = 1,25 \cdot 2,8 = 3,5 \text{ м}$$

Шырағдан мен қабырға арасындағы қажетті қашықтық:

$$l_A = 0,5 \cdot L_A = 0,5 \cdot 3 = 1,5 \text{ м}$$

$$l_B = 0,5 \cdot L_B = 0,5 \cdot 3,5 = 1,75 \text{ м}$$

Шамдарды және жарық ойықтарын орналастыру сұлбасы 5.3-суретте көрсетілген.



5.3-сурет – Шамдарды және жарық ойықтарын орналастыру сұлбасы

### 5.3 Өміртіршілік қауіпсіздігі бөлімі бойынша қорытынды

Жүргізілген есептеулер негізінде жасанды жарықтандыру қайта жаңартылды. Жарық ағыны 3570 лм, қуаты 65Вт ЛД65-7 G13 ескі люменцентті шамдарды ЛД80-7 G13 жарық ағыны 4250 лм, қуаты 80 Вт ЛД80-7 жаңа люменцентті шамдарға ауыстыру және шырағдандағы шамдардың санын көбейту керек деп шешілді.

## Қорытынды

Бұл дипломдық жұмысты орындау кезінде Университет түлектерінің бірлестігі веб-бағытталған жүйесін құруға арналған талаптар қарастырылды.

Веб-бағытталған жүйені құруға, оның дизайн және дизайн үлгілеріне, жүйелік жобалау әдістеріне, моделіне зерттеу және ғылыми-техникалық талдау жүргізілді. Жалпы теория жүйесіне негізделіп отырып жүйеге жобалау нысаны ретінде талдау жүргізілді.

Жұмысты орындау барысында жаңалықтар лентасын, жаңа пайдаланушыларды тіркеу және оларға роль беру мүмкіндігін, файлдарды, құжаттарды жүктеу және сақтау мүмкіндігін, сондай-ақ ақпаратты тез қосу және өзгерту мүмкіндігін қамтитын үш тілді веб-бағытталған жүйе әзірледім. Жүйе PHP-фреймворк Yii2 негізінде әзірленген, ол сайттың контентін басқарудың заманауи жүйелеріне қойылатын барлық талаптарға жауап береді және PhpMyAdmin мәліметтер қорын басқарушы қосымшасы және OpenServer желілік сервері мен MySQL деректер қоры пайдаланылды.

Php – скриптің модульдік жүйесі электронды жүйеге мен перспективаларды ашады, жаңа әзірлеудің орнына дайын технологияны пайдалану сайтты іске қосу мерзімін айтарлықтай қысқартады және жалпы құнын төмендетеді.

Дипломдық жұмыстың соңында әзірленген жүйе интерфейсі оның функционалдылығымен және пайдалану қарапайымдылығымен сәтті үйлеседі деп айтуға болады. Мамандарға жүгінбей-ақ, сайт мазмұнын өз бетінше жедел толықтыру және өзгерту мүмкіндігі іске асырылды.

Жүйе жұмысының қауіпсіздігі қауіпсіздіктің стандартты әдістерімен қамтамасыз етіледі (күрделі парольдерді құру, сақтау сенімділігі, сайт админінің парольдерін айына бір рет ауыстыру).

Жобаның экономикалық көрсеткіштері жыл бойы сайтты әзірлеу және игеру шығындарын өтеуге мүмкіндік береді. Сонымен қатар, өміртіршілік қауіпсіздігінің параметрлері программа әзірленген бөлмеде нормалар мен стандарттарға сәйкес келеді.

Жоғарыда айтылғандарды ескере отырып, жүйе осы жобаның ресурстарын басқару үшін ең оңтайлы болып табылады деген қорытынды жасауға болады. Зерттеу барысында жинақталған материалдарды талдау негізінде «Университет түлектерінің бірлестігі» ең тиімді деп таңдалған бағдарламасы құрылды.

## Әдебиеттер тізімі

- 1 Қазақстан Республикасы нормативтік құқықтық актілерінің ақпараттық-құқықтық жүйесі. [Электрондық ресурс]. URL: <http://adilet.zan.kz>
- 2 Уайтхэд, Пол Perl. Наглядный курс программирования / Уайтхэд Пол, Эрик Крамер. – М.: Диалектика, 2001. - 280 с.
- 3 Савитч, Уолтер Язык Java. Курс программирования / Уолтер Савитч. - М.: Вильямс, 2015. - 928 с.
- 4 Эспозито, Д. Программирование с использованием Microsoft ASP.NET 4 / Д. Эспозито. - М.: Питер, 2013. - 741 с.
- 5 Бардзелл, Джеффри Macromedia Dreamweaver MX 2004 с ASP, ColdFusion и PHP. Из первых рук (+ CD-ROM) / Джеффри Бардзелл. - М.: Эком, 2017. - 560 с.
- 6 Дронов, В.А. PHP 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов / В.А. Дронов. - М.: БХВ-Петербург, 2012. - 418 с.
- 7 Колисниченко, Денис Профессиональное программирование на PHP (+CD-ROM) / Денис Колисниченко. - М.: БХВ-Петербург, 2015. - 416 с.
- 8 Кузнецов, М. Объектно-ориентированное программирование на PHP / М. Кузнецов, И. Симдянов. - М.: БХВ-Петербург, 2012. - 608 с.
- 9 Гультияев, А. К. Уроки Web-мастера. Технология. Дизайн. Инструменты / А.К. Гультияев, В.А. Машин. - М.: Корона-Принт, 2016. - 448 с.
- 10 Корзина М.И. Дизайн-технология разработки интернет-сайтов // Диссертация на соискание ученой степени кандидата технических наук. - 2014. - 161 с.
- 11 Сафронов, Марк Разработка веб-приложений в Yii 2 / Марк Сафронов. - М.: ДМК Пресс, 2016. - 392 с.
- 12 Вандюк, Джон К. CMS Drupal. Руководство по разработке системы управления сайтом / Вандюк, Джон К., Мэтт Вестгейт,. - М.: Вильямс, 2016. - 400 с.
- 13 Семакин, И. Г. Основы программирования и баз данных. Учебник / И.Г. Семакин - М.: Академия, 2014. - 224 с.
- 14 Скляр, Дэвид PHP. Рецепты программирования / Дэвид Скляр, Адам Трахтенберг. - М.: Питер, 2017. - 784 с.
- 15 Харрис, Энди PHP/MySQL для начинающих / Энди Харрис. – М.: КУДИЦ-Образ, 2016. - 384 с.
- 16 Пауэрс, Дэвид PHP. Создание динамических страниц / Дэвид Пауэрс. - М.: Рид Групп, 2012. - 640 с.
- 17 Уайт, Э. PHP 5 на практике / Э. Уайт. - М.: ИТ Пресс, 2014. - 959 с.
- 18 Бенкен, Е. PHP, MySQL, XML. Программирование для Интернета / Е. Бенкен. - М.: БХВ-Петербург, 2016. - 352 с.
- 19 Локхарт, Джош Современный PHP. Новые возможности и передовой опыт / Джош Локхарт. - М.: ДМК Пресс, 2016. - 304 с.
- 20 Сырых Ю. А. Современный веб-дизайн. Эпоха Веб 3.0 / Ю. А. Сырых. – М.: Диалектика, 2013. - 368 с.

- 21 Хоган Б. Книга веб-программиста. Секреты профессиональной разработки веб-сайтов / Б. Хоган, К. Уоррен, М. Уэбер, К. Джонсон, А. Годин. - Питер - Москва, 2013. - 288 с.
- 22 Мюллер, Р. Дж. Базы данных и UML. Проектирование / Р. Дж. Мюллер. - М.: ЛОРИ, 2015. - 420 с.
- 23 Шаймарданов, Р. Б. Моделирование и автоматизация проектирования структур баз данных / Р. Б. Шаймарданов. - М.: Радио и связь, 2013. - 120 с.
- 24 Яргер, Р. Дж. MySQL и mSQL: Базы данных для небольших предприятий и Интернета / Р. Дж. Яргер, Дж. Риз, Т. Кинг. - М.: СПб: Символ-Плюс, 2014. - 560 с.
- 25 Китинг Flash MX. Искусство создания web-сайтов / Китинг, Джоди. - М.: ТИД ДС, 2014. - 848 с.
- 26 Томлинсон, Тодд CMS Drupal 7. Руководство по разработке системы управления веб-сайтом / Тодд Томлинсон. - М.: Вильямс, 2016. - 560 с.
- 27 Информационный портал Intellect. [Электрондық ресурс] – 2016. – URL: <https://intellect.icu/>
- 28 Дж. Гарретт. Элементы опыта взаимодействия. [Электрондық ресурс]. – 2008. – URL: <https://issuu.com/alextytarchuk/docs/garrett-web-design>
- 29 Официальный сайт Google Trends. [Электрондық ресурс]. – URL: <https://trends.google.com/trends>
- 30 Open Server – профессиональный инструмент веб-разработчика под Windows. [Электрондық ресурс]. – 2015. – URL: <https://ospanel.io/>



## **А қосымшасы** (міндетті)

### Техникалық тапсырма

#### **А.1 Кіріспе**

Веб-бағытталған жүйе құру интернет ресурстарын әзірлеудің маңызды технологияларының бірі. Жақсы ақпараттық жүйе өзіне барлық пайдалы деректерді жинай отырып, тәуліктің кез келген уақытында қарқынды жұмыс жасайды және коммерциялық фирманың немесе білім беру мекемесінің ең жақсы визит карточкасы болып табылады.

Бүгінгі күні интернетте саяхаттай отырып, тез жүктелетін, тартымды, бөлімдер бойынша нақты шектелген, ақпарат пен навигацияның ыңғайлы функциялары бар өте жақсы орындалған сайттарды табуға болады. Қолданушы мұндай сайттарда біраз уақыт бойы кідіріп, жаңа әрі қызықты ақпарат ала алады. Бұл дипломдық жоба әзірлеу барысында ескеретін маңызды мәселе.

Университет түлектерінің бірлестігіне арналған веб-бағытталған жүйені құру өзектілігі – Университет пен оның түлектері арасындағы өзара тиімді қарым-қатынасқа қол жеткізу, соның ішінде университет тарапынан бітіруші түлектердің кәсіби дамуының барлық кезеңдерінде қолдау көрсетуі және соған орай бітірушілердің университетті дамытуға, ғылыми және білім беру жобаларына ат салысуы болып табылады.

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

Дипломдық жобаның мақсаты – Университеттің түлектеріне арналған бірлестіктің веб-бағытталған жүйесін әзірлеу, студенттердің білімі туралы анықтама алуды автоматтандыру және ақпаратқа жылдам қол жеткізу үшін сенімді әрі сапалы жүйені құру болып табылады.

Университеттің түлектеріне арналған бірлестіктің веб-бағытталған жүйесінің мақсаты:

- түлектердің Университетпен және өзара қарым-қатынастарының үздіксіз байланысы;
- бірлестік ішіндегі және Университетпен кәсіби, шығармашылық және ғылыми байланыстарды дамыту;
- түлектер туралы ақпарат және сол ақпаратқа түлектердің қашықтықтан қол жеткізуі.

Дипломдық жобаның негізгі тапсырмалары:

- қолданыстағы жүйелерді зерттеу, талдау және сараптау;
- бағдарлама құру үшін нақты құрал-жабдықтарды таңдау;
- бағдарлама құру барысындағы экономикалық тиімділіктерді зерттеу;
- бағдарлама құру кезіндегі тіршілікке қолайлы ортаны зерттеу;

## *А қосымшасының жалғасы*

– бағдарламаны қолданысқа енгізу.

Жоба PHP скрипт тілінің құрамдас құрылымына негізделген Yii2 жоғары тиімді фреймворк, сондай-ақ MySQL ДҚБЖ көмегімен жүзеге асырылды.

### **А.1.2 Қолдану саласы**

Қолдану саласы – Жоғарғы оқу орны.

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

Анықтамалар, терминдер және қысқартулар А.1-кестеде көрсетілген.

А.1-кесте – Анықтамалар, терминдер және қысқартулар

Терминдер немесе қысқартулар	Анықтамалар
front-end әзірлеуші	Сайтты жүктеу жылдамдығына және кодтың дұрыс жұмыс жасалуына жауапты
back-end әзірлеуші	Сервердің кодын жазу үшін жауапты
full-stack әзірлеуші	Тұтынушының барлық талаптарына толық жауап бере алады.
PHP	Hypertext Preprocessor – гипертекст препроцессоры
COO	Concept of Operations – жүйенің тұжырымдамасы
LCO	Life Cycle Objectives өмірлік циклдің мақсаты мен мазмұны
LCA	Life Cycle Architecture – өмір циклінің архитектурасы, мұнда программалық жасақтаманың тұжырымдық архитектурасының дайындығы туралы айтуға болады
IOC	Initial Operational Capability – бастапқы операциялық мүмкіндіктер
FOC	Final Operational Capability – нақты жұмыс істеуге арналған (дайындалған және конфигурацияланған) дайын өнім
HTML	HyperText Markup Language – гипермәтіндік белгілеу тілі
MySQL	Деректер қорын басқарудың еркін реляциялық жүйесі

## **А.2 Жалпы сипаттамасы**

Түлек жүйеге өтініш қалдырады, админ Университетте оқығандығын базадан тексереді, расталса жүйеге тіркейді. Түлек логин, паролін алып, жеке кабинетке кіре алады және анықтама, құжаттарды сақтайды. Админ барлық мәліметтерді, қолданушыларды және сайт беттерін басқарады. Менеджер тек түлекке қатысты мәліметтерді ғана өзгертеді, оқиға немесе іс-шара енгізу арқылы түлектерге шақырту жібереді. Барлық мәліметтерді Excel және PDF-ке түрлендіруге болады.

### **А.2.1 Пайдаланушылық интерфейстер**

Сайттың оң жақ шетінде «Бірлестікке кіру» батырмасы орналасқан. Ол түлектердің бірлестікке кіруге өтініш тастауына мүмкіндік береді.

«Бірлестікке кіру» батырмасы РНР тілінде форма ретінде жасалған және батырманы басқанда сол бетте қалқыған терезе пайда болады. Жіберу батырмасын басқан кезде жауап беріледі.

Ал жіберілген хабарламалар админ электронды почтасына келіп түседі. Бұл админ почтасын енгізу көрсетілген жолдағы программа кодында көрсетілген: /config/params.php. Орталықтың почтасына кіруге кімде рұқсат бар, сол қызметкерлер келген хабарламаларды оқып жауап қайтара алады.

Админ бөліміне кіру үшін «Кіру» батырмасын басамыз және ол арнайы логин және құпия сөзді енгізу арқылы орындалады. Админ құпия сөзі деректер қорында шифрланған түрде сақталған. Егер құпия сөз деректер қорындағы құпия сөзбен сәйкес келмесе, онда құпия сөздің дұрыс еместігі туралы хабар беріліп, қайта енгізу мүмкіндігі беріледі.

Админ логині және құпия сөзін енгізгеннен кейін, ол автоматты түрде админ бөліміне енеді және мәліметтерді реттейді. Админ бөлімі конфигурация, панель, институт/факультет, мамандықты басқару, түлектер, жинақ, статистика, қолданушылар құқығы бөлімдерінен тұрады.

Конфигурация бөлімінде Университет, мекен-жай, мерекелер, статус жөнінде ақпарат орналасқан, оны админ басқарып отырады. Сондай-ақ, түлекке қажетті құжаттар санатын енгізуге болады, сол құжаттардың санатына қарай түлектер өзіне қажетті анықтамаларды өзінің жеке кабинетінен сақтай алады. Облыс, қала, аудан, тілдер, статус түлек немесе менеджерді қосқан кезде қажетті ақпарат болып табылады.

Админ логині және құпия сөзін енгізгеннен кейін, ол автоматты түрде админ бөліміне енеді және мәліметтерді реттейді. Админ бөлімі конфигурация, панель, институт/факультет, мамандықты басқару, түлектер, жинақ, статистика, қолданушылар құқығы бөлімдерінен тұрады.

Конфигурация бөлімінде Университет, мекен-жай, мерекелер, статус жөнінде ақпарат орналасқан, оны админ басқарып отырады. Сондай-ақ,

түлекке қажетті құжаттар санатын енгізуге болады, сол құжаттардың санатына қарай түлектер өзіне қажетті анықтамаларды өзінің жеке кабинетінен сақтай алады. Облыс, қала, аудан, тілдер, статус түлек немесе менеджерді қосқан кезде қажетті ақпарат болып табылады.

Панель бөлімінде әр қолданушының панеліне арнайы хабарламалар, немесе ескертпе қалдыруға болады және күнтізбеге оқиға енгізу арқылы түлектерді кездесулерге, конференцияларға немесе Университетте өтетін кеш қонағы ретінде шақыруға болады.

Жүйеде түлектерді тіркеу үшін, олар туралы мәлімет енгізу керек. Яғни, аты-жөні, оқыған мамандығы, факультеті, қазіргі мекен-жайы, жұмыс орны және тағы сол сияқты ақпаратты енгізу қажет. Бұл жерде бітіру жылын таңдағанда сол жылғы институт немесе факультет көрсетіледі, ал институт немесе факультетке қарай мамандықтарды таңдауға болады және оны басқа кестемен байланыс жасайтын кестенің сыртқы кілті FOREIGN KEY арқылы жүзеге асырдым

Түлек жеке кабинетіндегі құжаттарды өзіне сақтай алады және өз жеке кабинетінде барлық ақпаратты көре алады, бірақ барлығын бірдей өзгерте алмайды, себебі ондай құқық тек админ, менеджерде ғана. Алайда профильдегі фотоны өзгерте алады, ол үшін көрсетілген батырманы басамыз.

### **A.2.2 Аппараттық интерфейстер**

Компьютерлерге қойылатын жалпы талаптар:

- жоғары процессор;
- 256 МВ жоғары оперативті жады;
- диск тұлғалы кеңістік – 52 МВ қолданушылар файлдарын сақтауға қажет орын;
- интернет желісіне арналған қамтама;

### **A.2.3 Программалық интерфейстер**

Серверлік программалы компоненттер:

- кез келген операциялық жүйе;
- MySQL ДҚБЖ;
- Yii2 фреймворк тәуелділіктерін орнату;

Клиенттік программалы компоненттер:

- кез келген операциялық жүйе;
- Google Chrome, Internet Explorer, Safari және т. б.

### **A.2.4 Коммуникациялық интерфейстер**

Пайдаланушылардың компьютерлерінде интернетке қатынауға бар

модем құралдары немесе жылдамдығы 100 Мб/с кем емес желілі карта болуы тиіс. Сервер мен клиенттерді байланыстыратын TCP/IP протоколы қолданылады.

### **А.2.5 Жады бойынша шектеулер**

Бағдарлама үшін жалпы жады бойынша қойылатын талаптар, олардағы жұмысқа қажетті программалық компоненттердің қойылатын талаптарымен анықталады.

### **А.2.6 Адаптация бойынша талаптар**

Программалық кешен кез келген операциялық жүйелерде, яғни Win32, Solaris, HP/UX, Linux және Windows платформаларында жұмыс істеуі тиіс, қажетті программалық компоненттер ыңғайлы орнатылуы керек.

## **А.3 Өңдеуге талаптар**

### **А.3.1 Функциональды талаптар**

Жүйеге кіру үшін әрбір админ, менеджер және түлек өзінің логині және паролі арқылы кіруі керек.

#### **А.3.1.1 Шығыс деректері**

Деректер қорына кіретін мәліметтер ретінде, менеджер, түлек туралы ақпарат, оның аты-жөні, электрондық поштасы, құпия сөзі жатады.

Админ жағынан барлық мәлімет экспортталады.

#### **А.3.1.2 Нәтижелер**

MySQL деректер қорында админ, менеджер және түлек туралы ақпарат жауаптары сақталынады.

#### **А.3.1.3 Функциональды талаптарды ұйымдастыру**

Бағдарлама арқылы келесідей функцияларға қол жеткізеді:

– түлектер туралы, соның ішінде, байланыс деректері, мекен-жайы, қазіргі жағдайы және жұмыс орны туралы ақпарат;

– анықтамалық ақпарат, яғни Университетте оқығаны жайлы анықтама, диплом және т.б.;

– хабарлама жіберу арқылы қашықтықтан тіркелу, жүйеге кіру;

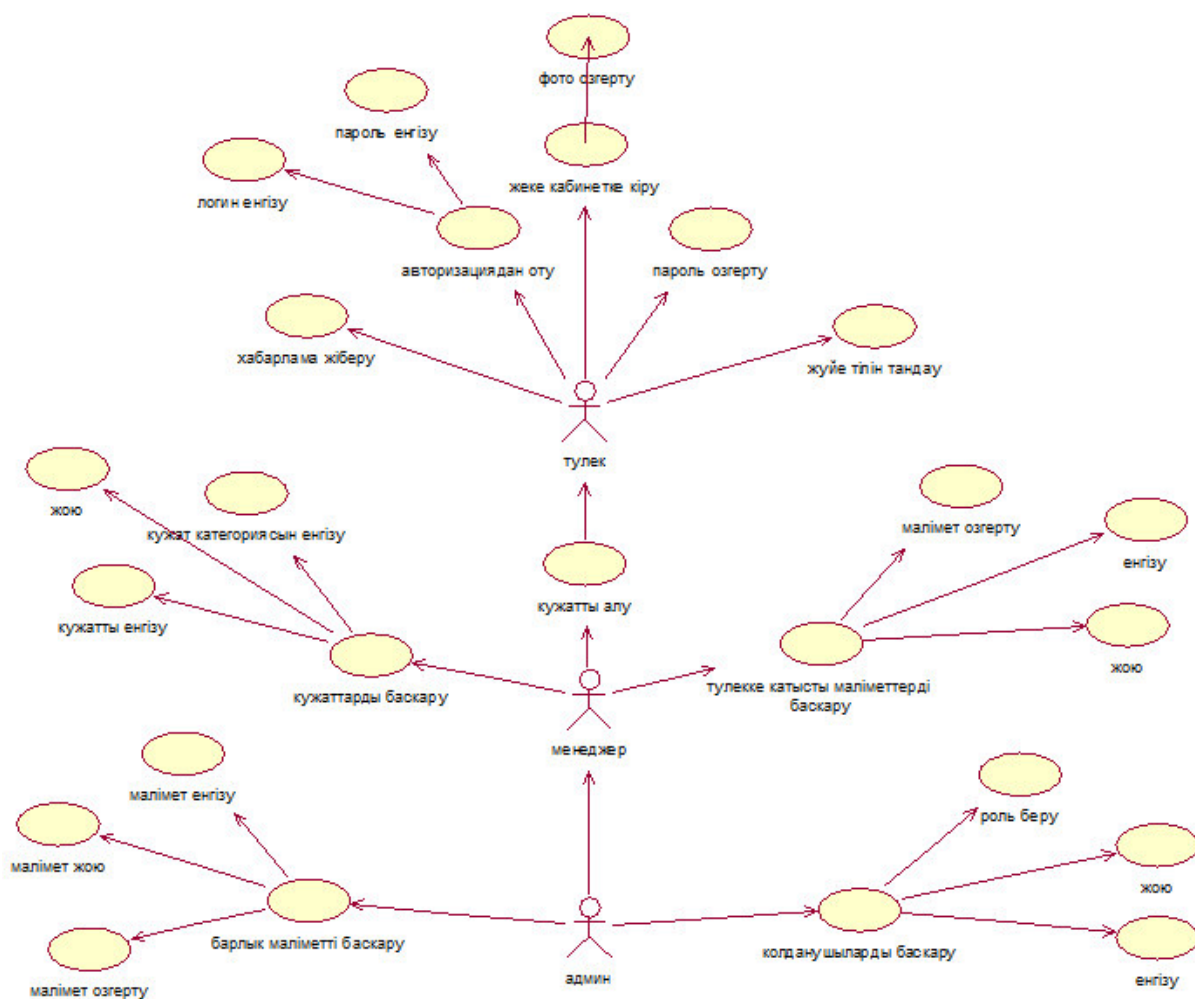
### А қосымшасының жалғасы

- Университет жайлы соңғы әрі өзекті ақпарат;
- түлектер кездесуі, маңызды іс-шараларға шақырту алу;
- Университетке инвесторлық көмек.

#### А.3.1.3.1 Функционалды диаграммалар

«Университет түлектерінің бірлестігі» веб-бағытталған жүйесі үш жақты қолданушыларға негізделген: админ, менеджер және түлек.

3.1-суретте көрсетілген диаграмма қолданушылар орындай алатын іс-әрекеттерді көрсетіп тұр, яғни админ барлық іс-әрекеттерді басқарады, соның ішінде қолданушыларды, мәліметтерді, құжаттарды және т.б. Ал менеджер болса, түлекке қатысты мәліметтер мен құжаттарды ғана өзгерте алады және түлек секілді жеке кабинетке кіреді, пароль өзгертеді, хабарлама жібереді және т.с.с.

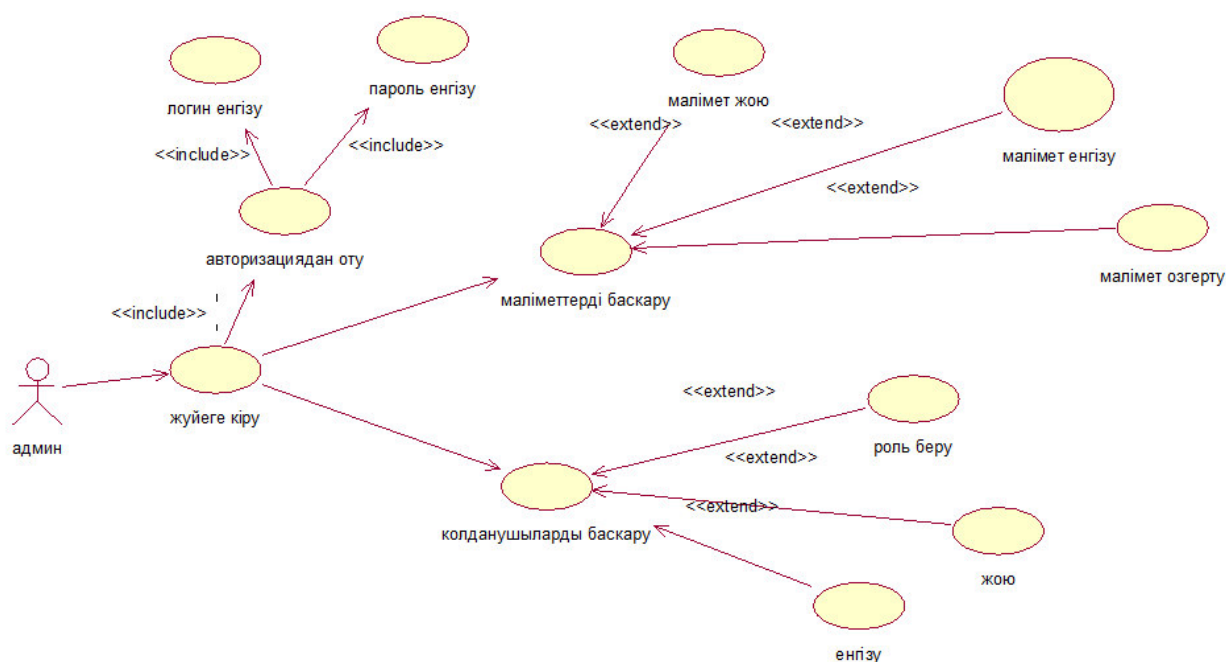


3.1-сурет – Жүйенің қолдану нұсқаларының жалпыланған диаграммасы

### А қосымшасының жалғасы

Админ – пайдаланушының рөлі. Сондай-ақ, админ басқа пайдаланушыларға сайт админі рөлін тағайындай немесе жоя алады (3.2-сурет). Админ рөлін басқа пайдаланушыларға тағайындағанда, көрсетілген рөлмен сіз оларды сайттан мазмұнды алып тастау мүмкіндігіңіз бар екенін есте сақтау өте маңызды, сондықтан өте сақ болу керек.

Тек админ сайтты жаңарта алады. Мысалы, ол тақырыптарды өзгертіп, негізгі файлдарды кірістірілген тақырып редакторымен өңдей алады. Мұнда админ сайтта кез келген плагиндерді қосуға, жоюға және өзгертуге құқылы.



3.2-сурет – Админ диаграммасы

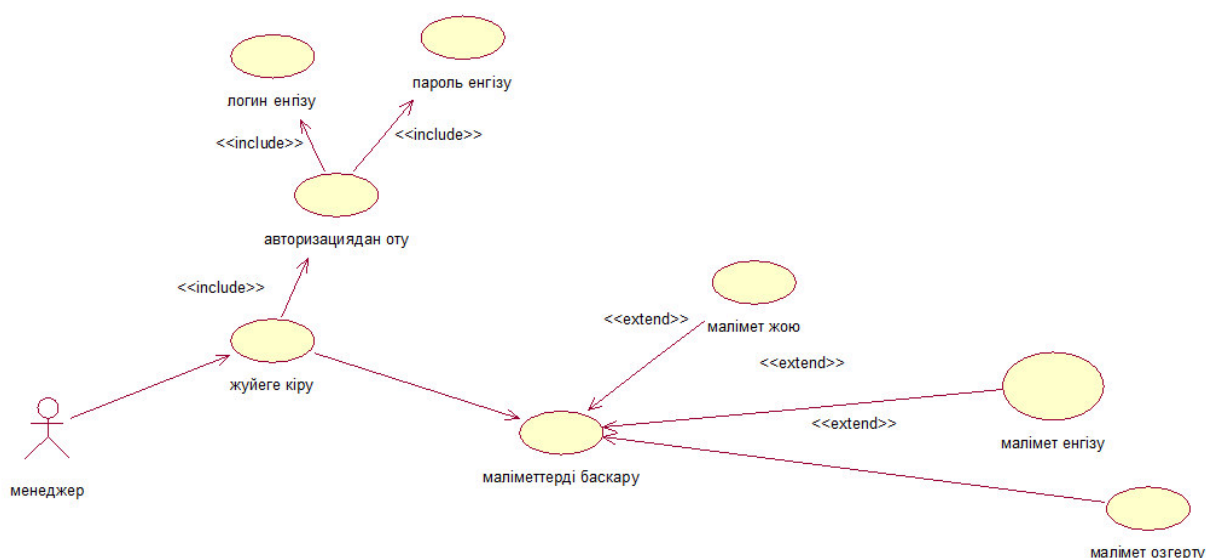
Сайтты баптармен толтыру ақпараттың сапасы, оның оқу және қабылдау жеңілдігі, өзектілігі және пайдаланушының сұрау салуына сәйкестігі маңызды болып табылады.

Әр түрлі сайттарға әртүрлі ақпараттық мазмұн талап етіледі, сондықтан менеджер сайттың ерекшеліктерін түсінуі тиіс.

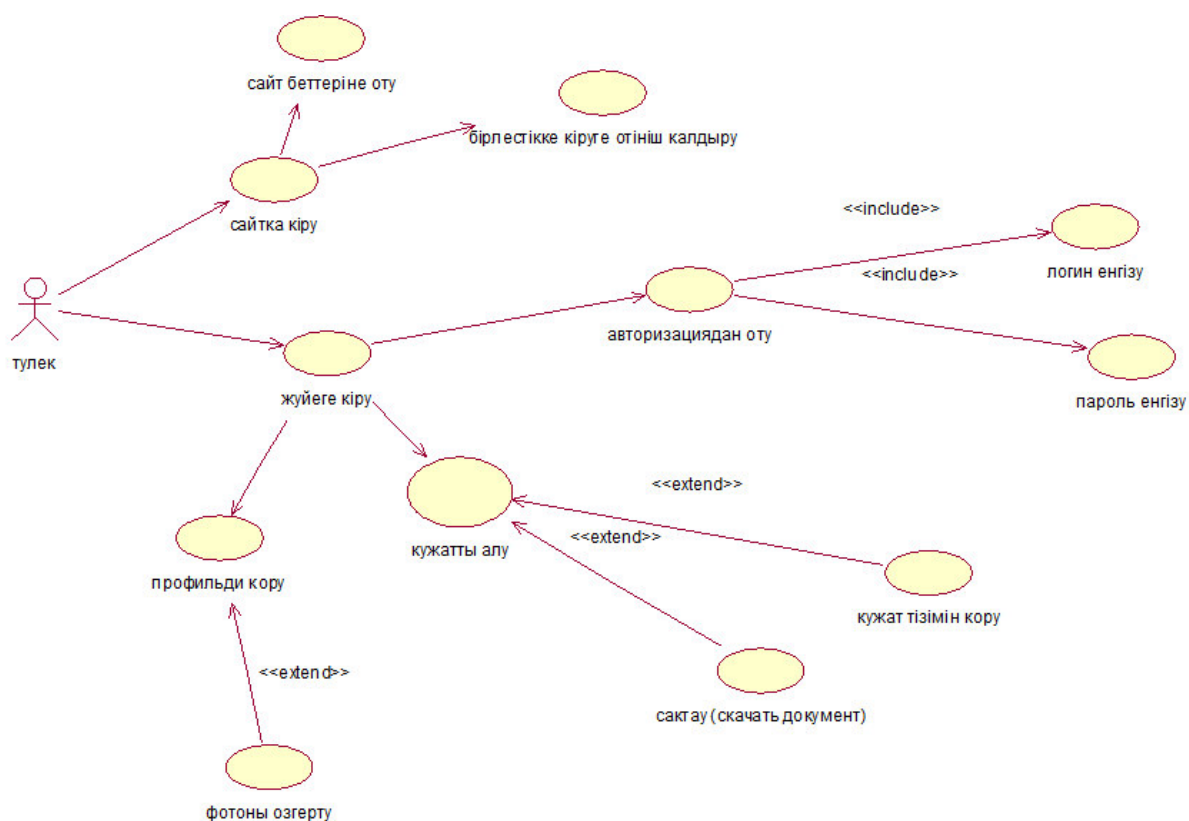
Менеджер өзіне сеніп тапсырылған сайтта басқаруды толық меңгеруі, оны өзекті және жұмысқа қабілетті күйде ұстауы тиіс.

Бұл жүйедегі менеджер мәліметтерді ғана басқарып қоймай, түлектерге хабарламалар, шақыртулар жән ескертпелер жіберіп отырады. Кез келген кафедра немесе деканат қызметкері бұл рөлге ие бола алады, админ арқылы жүйеге тіркеледі. Менеджер іс-әрекеті 3.3-суретте көрсетілген.

## А қосымшасының жалғасы



3.3-сурет – Менеджер диаграммасы



3.4-сурет – Түлек диаграммасы

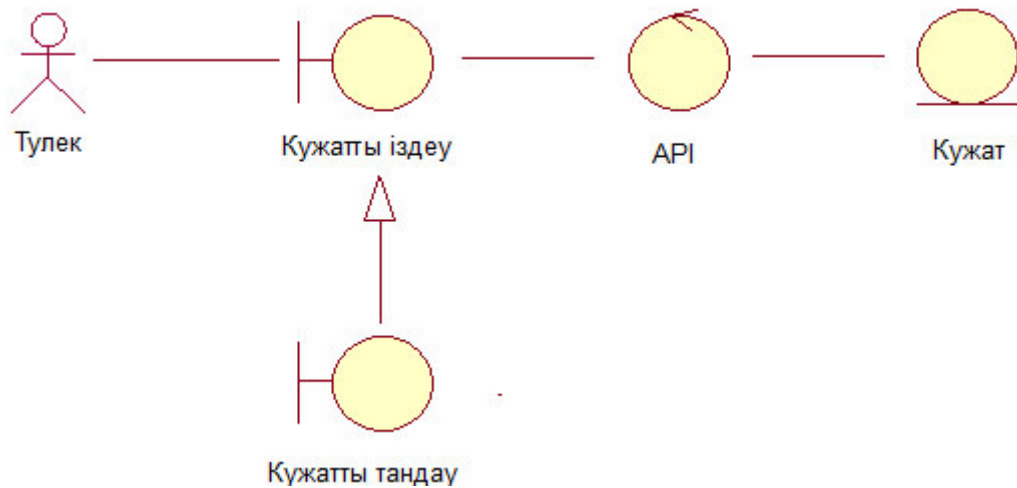
Түлектке қажетті құжат, анықтамаларды менеджер жеке кабинетке салады, процесс қалай жүретіні 3.4-суретте көрсетілген. Сәйкесінше түлек сол құжаттарды өз жеке кабинетіне кіру арқылы көре алады және өзіне тиесілі әрекеттерді жасайды (3.5-сурет).



### А қосымшасының жалғасы



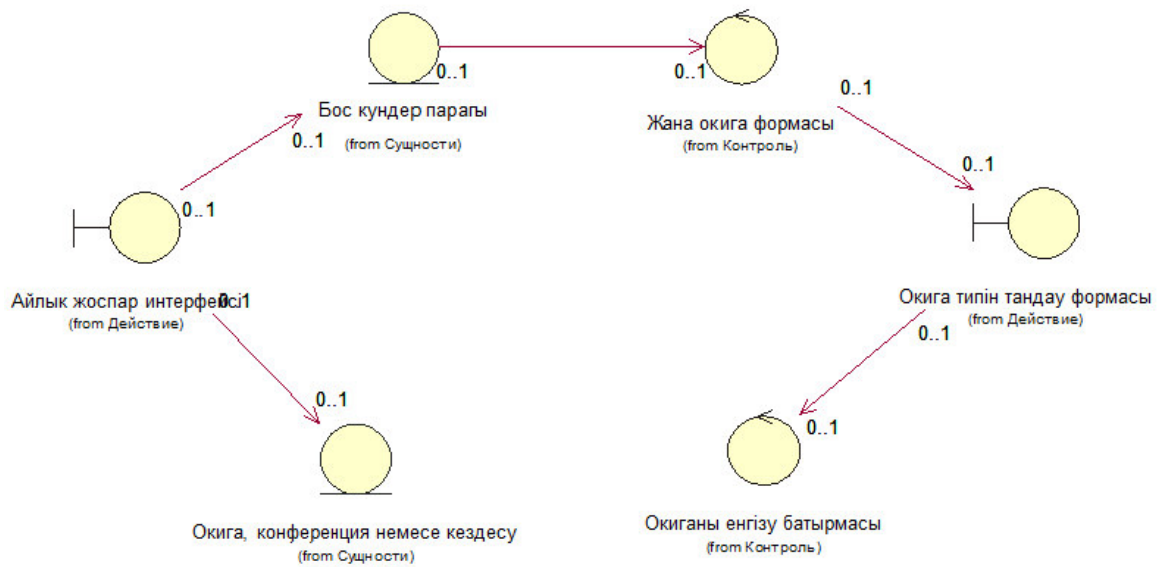
2.9-сурет – «Кұжатты салу» анализ диаграммасы



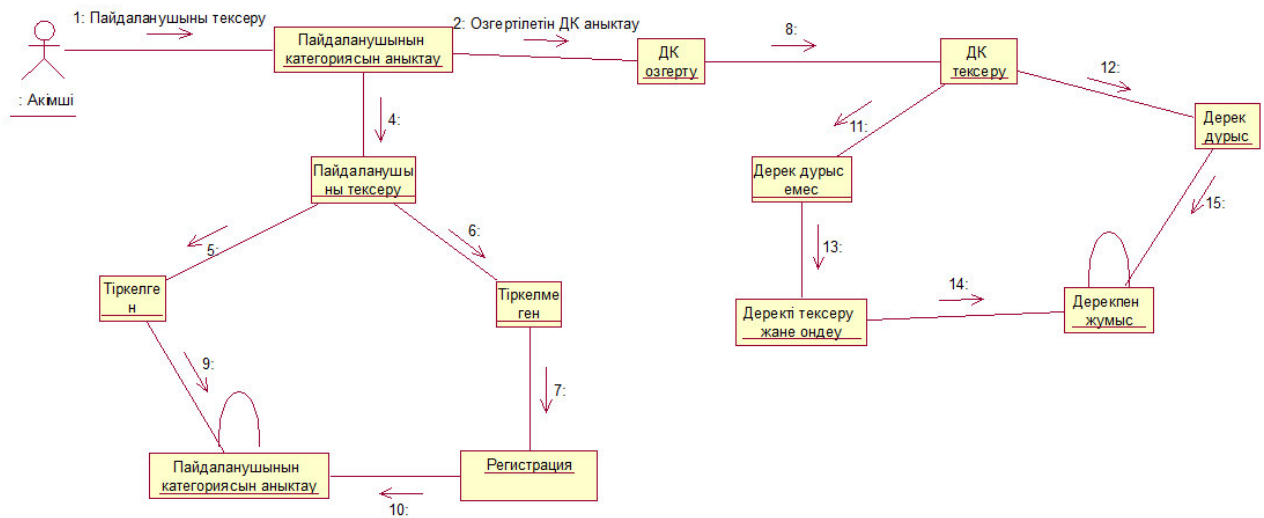
3.5-сурет – «Кұжатты алу» анализ диаграммасы

Логикалық жоспарлау диаграммасына мысал ретінде оқиға енгізуді қарастырдым (3.6-сурет). Түлектерді кездесуге немесе іс-шараларға шақыруға болады, ол админ және менеджер арқылы жүзеге асырылады. Яғни, оқиға уақыты, өтетін орны туралы мәліметтер енгізіп, қолданушы типіне түлектер деп белгілейміз де жібереміз. Түлек өз жеке кабинетіне кірген кезде енгізілген оқиға көрінеді.

### А қосымшасының жалғасы



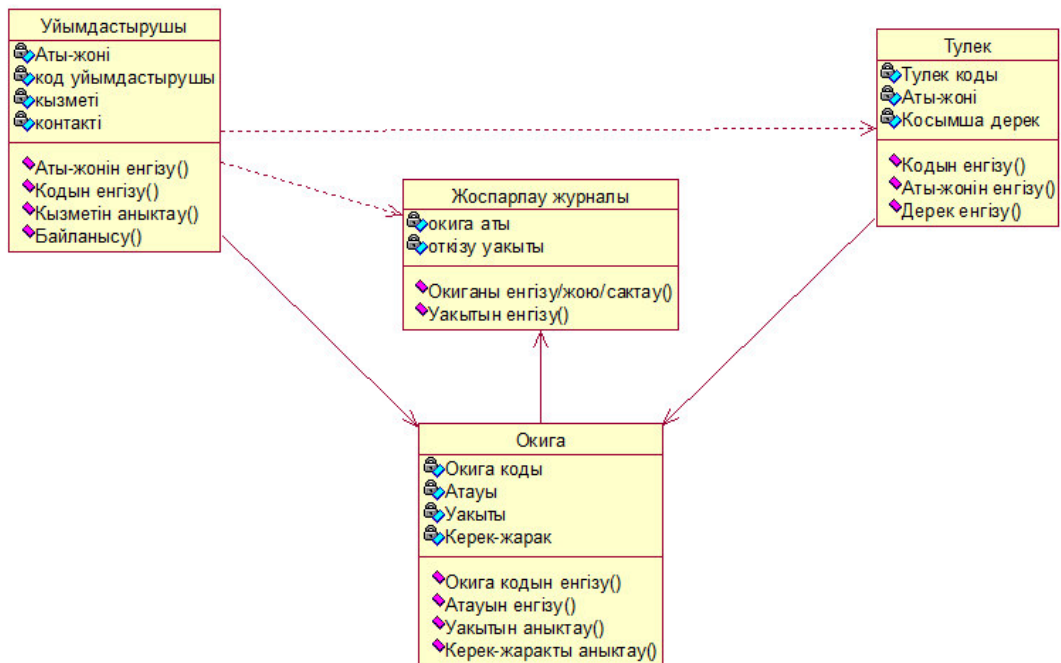
3.6-сурет – Оқиға енгізудің логикалық жоспарлау диаграммасы



3.7-сурет – Логикалық диаграмма

UML тіліндегі класс басқа құрылымдардағы объекттермен бірдей құрылымға, мінез-құлыққа және қарым-қатынастарға ие объектілер жиынтығын белгілеу үшін пайдаланылады. Оқиға енгізудің класстар диаграммасы 3.8-суретте көрсетілген.

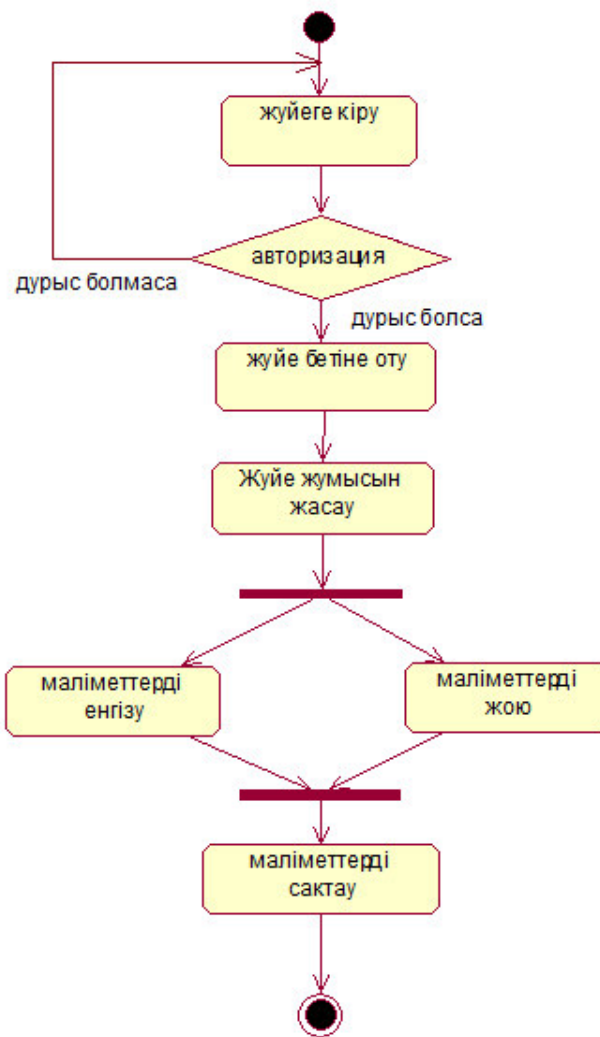
## А қосымшасының жалғасы



3.8-сурет – Оқиға енгізудің класстар диаграммасы

Процесс бастапқы нүктеден басталады, содан кейін "Жүйеге кіру" күйіне ең бірінші өту керек. Жүйедегі объектінің мінез-құлқына ауысулар мен күйлер көрсететін әрекеттерді жатқызуға болады. Басқа болса да, бұл "Жүйе" класының кейбір әдісімен жүзеге асырылатын процестер әртүрлі жолмен түсіндіріледі. Іс-қимылдар ауысумен байланысты және жылдам әрі еңсерілмейтін ретінде қарастырылады. Қызмет жағдайларға байланысты жеткілікті ұзақ уақытқа созылуы мүмкін немесе белгілі бір оқиғаға байланысты әрекетті тоқтатуға болады.

### А қосымшасының жалғасы



3.9-сурет – Жүйе админінің қызмет диаграммасы

Осылайша, объектілі-бағытталған талдау және күрделі жүйелерді жобалау процесі қызметтің диаграммасын қоса алғанда, жекелеген диаграммаларды төмендеу және өрлемелі әзірлеудің бірізділігі ретінде ұсынылады. Әзірлеудің қандай да бір бағыттарының Үстемдігі нақты жобаның ерекшеліктерімен және оның жаңалығымен айқындалады.

#### А.3.2 Функциональды емес талаптар

Жіберілген өтініш бойынша жауап алу үшін уақыт керек.

##### А.3.2.1 Өнімділігі

Жүйе сервердің максималды жүктелуі кезінде және кез-келген операциялық жүйелерде тиімді жұмыс істеуі тиіс. Веб-беттердің ауысуы тез орындалуы тиіс.

### **А.3.2.2 Қатынау және сенімділік**

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

Жүйе бірнеше клиенттік түйіндер қосылған кезде тиімді жұмыс істеуі тиіс және клиенттің компьютері мен деректер қорының синхронизациясын қамтамасыз етуі керек.

Жүйе келесі әрекеттерді тиімді орындауы тиіс:

- пайдаланушының кез келген әрекеттеріне қатесіз жауап беру;
- бірнеше клиенттік тораптарды қосу кезінде жұмыс істеу;
- синхрондауды қамтамасыз ету.

### **А.3.2.3 Қатені өңдеу**

Жүйедегі барлық қателер өңделуі тиіс және қателер туралы хабарлар пайдаланушылар қабылдай алатын түрде көрсетілуі керек. Хабарлау жүйесінде пайдаланушылардың мүмкін емес әрекеттерін, ақпарат енгізудің қате үлгілерін іске асыру керек.

### **А.3.2.4 Интерфейсті талаптар**

Қолданба веб-беттер арқылы админ, менеджермен өзара әрекеттеседі. Жүйеге кіру веб-интерфейс арқылы жүзеге асады.

### **А.3.2.5 Шектеулер**

Бағдарлама дербес компьютерлермен қолдайтын құралдармен іске асырылуы және қосымша техникалық құралдарды пайдалану үшін талап етпеуі тиіс.

### **А.3.3 Кері талаптар**

Деректер қорына қосылған кезде қордың атын, пайдаланушының атын және парольді дұрыс көрсету керек.

## Ә қосымшасы (міндетті)

### Программа листингі

```
<?php

use yii\helpers\Html;
use yii\bootstrap\Nav;
use yii\bootstrap\NavBar;
use yii\widgets\Breadcrumbs;
use app\assets\AppAsset;
use app\assets\ltAppAsset;
use yii\helpers\Url;
use app\components\MenuWidget;
use app\components\BlogWidget;
use app\components\LangWidget;

AppAsset::register($this);
ltAppAsset::register($this);
?>
<?php $this->beginPage() ?>
<!DOCTYPE html>
<html>
<html lang="<?= Yii::$app->language ?>">
<head>
<link rel="shortcut icon" href="/img/logo.png" type="image/jpg">
<meta charset="<?= Yii::$app->charset ?>">
<meta name="viewport" content="width=device-width, initial-scale=1">
<?= Html::csrfMetaTags() ?>
<title><?= Html::encode($this->title) ?></title>

<?php $this->head() ?>

</head>
<body>
<?php $this->beginBody() ?>
<!--preloader-->
<div id="preloader"></div>
<!--preloader-->
<!--back to top-->
<a href="#" class="scrollToTop"><i class="ion-android-arrow-dropup-circle"></i></a>
<!--back to top end-->
<header class="header-main">
<div class="top-bar">
<div class="container">
<ul class="info list-inline pull-left hidden-xs">
<li><a href="#" class="hover-color"><i class="ion-ios-email-outline">
</i>assotsatsiya.vypusknikov@mail.ru</a></li>
<li><i class="ion-iphone"></i> +7705 507 84 17</li>
```

```
</ul>
<div class="pull-right">
<ul class="info list-inline hidden-xs">
<li><a href="#" class="btn btn-success signup" style = "background-color:blue; color: white;"
data-userid="22"><?=Yii::t('app','Вступить в сообщество')?></a></li>
<li><a href="http://assotsiatsiya-vypusknikov" style = "background-color:blue; color: white;"
class="btn btn-success "><?=Yii::t('app','Войти')?></a></li>
<li>
<?= LangWidget::widget();?>
</li>
</ul>
</div>
</div><!--container end-->
</div><!--topbar end-->
<div class="navbar navbar-default yamm" role="navigation">
<div class="container">
<div class="navbar-header">
<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-
collapse">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
<a class="navbar-brand" href="<?=Url::to(['/'])?>"></a>
</div>
<div class="navbar-collapse collapse">
<ul class="nav navbar-nav navbar-right">
<?php echo MenuWidget::widget(['tpl'=>'menu']); ?>
<li class="social-nav"><a href="<?= Url::to('https://vk.com/away.php');?>"><i class="ion-social-
facebook"></i></a></li>
<li class="social-nav"><a href="<?= Url::to('https://mail.ru');?>"><i class="ion-
email"></i></a></li>
</ul>
</div><!--/.nav-collapse -->
</div><!--/.container-->
</div><!--navigation end-->
</header><!--header main end-->

<?=$content?>

<div id="footer">
<div class="container">
<div class="row">
<div class="col-md-4 margin-btm-20">
<div class="footer-col">
<h3><?=Yii::t('app','Ассоциация выпускников')?></h3>
<p>
```

## Ә қосымшасының жалғасы

```
<?=Yii::t('app','Дистанционный доступ выпускников к информации')?>
</p>
</div>
</div><!--col-4 end-->
<div class="col-md-3 margin-btm-20">
<div class="footer-col">
<h3><?=Yii::t('app','Контактные данные')?></h3>
<p><i class="ion-home"></i><?=Yii::t('app',' А.Байтурсынова, 126/1')?></p>
<p><i class="ion-iphone"></i> +7705 507 84 17</p>
<p><i class="ion-email"></i> assotsatsiya.vypusknikov@mail.ru</p>
</div>
</div><!--latest post col end-->
<div class="col-md-5">
<div class="footer-col">
<h3><?=Yii::t('app','Подписывайтесь на нас')?></h3>
<ul class=" list-inline social-btn">
<li><a href="<?= Url::to('https://www.facebook.com/groups/999389116921029/');?>"><i
class="ion-social-facebook" data-toggle="tooltip" data-placement="top" title="" data-original-
title="Like On Facebook"></i></a></li>
<li><a href="#"><i class="ion-social-twitter" data-toggle="tooltip" data-placement="top" title=""
data-original-title="Follow On twitter"></i></a></li>
<li><a href="#"><i class="ion-social-googleplus" data-toggle="tooltip" data-placement="top"
title="" data-original-title="Follow On googleplus"></i></a></li>
<li><a href="#"><i class="ion-social-pinterest" data-toggle="tooltip" data-placement="top"
title="" data-original-title="pinterest"></i></a></li>
<li><a href="#"><i class="ion-social-skype" data-toggle="tooltip" data-placement="top" title=""
data-original-title="skype"></i></a></li>
</ul>
</div>
</div><!--get in touch col end-->
</div><!--footer main row end-->
</div><!--container-->
</div><!--footer main end-->
<!--footer main end-->

<!-- Modal "Вступить в сообщество" -->
<div class="modal fade" id="my-modal" tabindex="-1" role="dialog" aria-
labelledby="myModalLabel" aria-hidden="true">
<div class="modal-dialog modal-md">
<div class="modal-content">
<div class="modal-body">
...
</div>
</div><!-- /.modal-content -->
</div><!-- /.modal-dialog -->
</div><!-- /.modal -->
<?php $this->endBody() ?>
</body>
</html>
```



## Ә қосымшасының жалғасы

Түлектердің өтініш қалдыру моделінің коды:

```
<?php
namespace app\models;

use yii\base\Model;

/**
 * RegisterForm
 */
class RegisterForm extends Model
{
    public $name;
    public $phone;
    public $email;
    public $year;
    public $faculty;
    public $spec;
    public $message;
}

/**
 * Здесь мы указываем правила валидации - они могут быть как стандартными, так и
 * кастомными
 */
public function rules()
{
    return [
        ['name', 'required'],
        [['phone', 'email', 'year', 'faculty', 'spec', 'message'], 'safe'],
        [['email'], 'email'],
        [['phone', 'email'], 'validatePhoneEmailEmpty', 'skipOnEmpty'=> false],
    ];
}

public function attributeLabels()
{
    return [
        'name' => 'ФИО',
        'email' => 'Email',
        'phone' => 'Телефон',
        'year' => 'Год выпуска',
        'faculty' => 'Институт/Факультет',
        'spec' => 'Специальность',
        'message' => 'Сообщение',
    ];
}

/** Кастомная функция для проверки, что хотя бы одно из полей email или phone посетитель
 * заполнил */
public function validatePhoneEmailEmpty()
```

## Ә қосымшасының жалғасы

```
{
if(empty($this->phone) && empty($this->email))
{
$errorMsg= 'Укажите ваш email или телефон';
$this->addError('phone',$errorMsg);
$this->addError('email',$errorMsg);
}
if(!empty($this->phone) && (strlen($this->phone)<7))
{
$errorMsg= 'Слишком мало цифр в номере телефона';
$this->addError('phone',$errorMsg);
}
}
```

Түлектердің өтініш қалдыруының қалқыма терезесінің көрінісі:

```
<?php
```

```
use yii\helpers\Html;
use yii\bootstrap\ActiveForm;
```

```
$this->title = 'Стать членом ассоциаций';
$this->params['breadcrumbs'][] = $this->title;
?>
```

```
<div class="site-register container" >
<h1><?= Html::encode($this->title) ?></h1>
```

```
<p>Чтобы вступить в сообщество заполните и отправьте:</p>
```

```
<div class="row">
<div class="col-lg-8">
<?php $form = ActiveForm::begin(['id' => 'contactform']); ?>
<div class="row">
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12 controls">
<?= $form->field($model, 'name')->textInput(['maxlength' => 255], ['class' => 'input-modal']) ?>
</div>
</div></div>
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12 controls">
<?= $form->field($model, 'phone')->textInput(['maxlength' => 255], ['class' => 'input-modal']) ?>
</div>
</div>
</div>
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12">
```

## Ә қосымшасының жалғасы

```
<?= $form->field($model, 'email')->textInput(['maxlength' => 255], ['class' => 'input-modal']) ?>
</div>
</div>
</div>
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12">
<?php
$items=['Выбрать'=>'Выбрать','2014'=>'2014','2015'=>'2015','2016'=>'2016','2017'=>'2017','2018'=>'2018','2019'=>'2019'];
?>
<?= $form->field($model, 'year')->dropdownList($items) ?>
</div>
</div>
</div>
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12">
<?php
$items=['Выбрать'=>'Выбрать','ИСУИиИТ'=>'ИСУИиИТ','ИЭиЭ'=>'ИЭиЭ','ИТиТ'=>'ИТиТ','ИКИиТ'=>'ИКИиТ','ФАИТ'=>'ФАИТ','ФРТС'=>'ФРТС','ЭЭФ'=>'ЭЭФ','ТЭФ'=>'ТЭФ'];
?>
<?= $form->field($model, 'faculty')->dropdownList($items) ?>
</div>
</div></div>
<div class="col-md-6">
<div class="row control-group">
<div class="form-group col-xs-12">
<?php
$items=['Выбрать'=>'Выбрать','ВТиПО'=>'ВТиПО', 'ИС'=>'ИС','СИБ'=>'СИБ','ПС'=>'ПС', 'ТЭ'=>'ТЭ', 'ЭЭ'=>'ЭЭ', 'ЭСХ'=>'ЭСХ', 'РЭТ'=>'РЭТ'];
?>
<?= $form->field($model, 'spec')->dropdownList($items) ?>
</div>
</div></div>
<div class="col-md-12">
<div class="row control-group">
<div class="form-group col-xs-12 controls">
<?= $form->field($model, 'message')->textArea(['row' => 3], ['class' => 'input-modal']) ?>
</div>
</div>
</div></div>
<br>
<div id="success"></div>
<div class="row">
<div class="form-group col-xs-12 text-right" style="margin-top: -15px;">
<?= Html::submitButton(Yii::t('app','Отправить'), ['class' => 'btn theme-btn-color btn-lg'])?>
</div></div>
<?php ActiveForm::end(); ?></div></div></div>
```