ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

ҚОРҒАУҒА ЖІБЕРІЛДІ
Кафедра меңгерушісі
тех. ғыл. кандидаты,
ассистент-профессор
_____ Р. Юнусов
"20" мая 2019 ж.

Дипломдық жобаға
ТҮСІНІКТЕМЕЛІК ЖАЗБА

«Оқу пәнінің электронды каталогы» web-қосымшасы

5В060200 – «Информатика» мамандығы

Орындаған Мұсабекова Б.С.

Ғылыми жетекші
техн. ғыл. магистрі, лектор
_____ Қ. Марғұлан
"17" 05 2019 ж.

Алматы 2019

Отчет подобия

ҚАЗ ҰТУ

Университет:	Satbayev University
Название:	Мұсабекова Ботагез Инф(Б)-15.1К.docx
Автор:	Мұсабекова Ботагез
Координатор:	Қабылжан Маргұлан
Дата отчета:	2019-05-06 12:27:10
Коэффициент подобия № 1: ?	0,1%
Коэффициент подобия № 2: ?	0,0%
Длина фразы для коэффициента подобия № 2: ?	25
Количество слов:	5 021
Число знаков:	42 135
Адреса пропущенные при проверке:	
Количество завершенных проверок: ?	45

Ғылыми жетекші: _____ Қ. Маргұлан

Обоснование:

.....
.....
.....
.....
.....
.....

17.05.2019г ..... Мартулан К.

Дата *Подпись Научного руководителя*

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

5В060200 – Информатика мамандығы

Мұсабекова Ботагөз Сәбитқызы

Дипломдық жоба

Тақырыбы: «Оқу пәнінің электронды каталогы» web-қосымшасы

ҒЫЛЫМИ ЖЕТЕКШІНІҢ ПІКІРІ

Берілген дипломдық жобада студенттердің мамандық бойынша академиялық жоспарға сай жеке оқу жоспарын міндетті және элективті сабақтарды таңдау арқылы жеке оқу жоспарын бекіту мүмкіндіктерін беретін, сонымен қатар академиялық қарыз курстарын электронды түрде тіркейтін автоматтандырылған «Оқу пәнінің электронды каталогы» жүйесін ұйымдастыру барысы қарастырылған.

Дипломдық жобаны Мұсабекова Ботагөз Сәбитқызы жоспарлы түрде, қойылған талаптарға сай, оңтайлы нәтижелер алып, жұмысты қойылған талаптарға сай орындап, тақырып төңірегінде жазып шыққанын ескере отырып, Мұсабекова Ботагөз Сәбитқызына Информатика – 5В060200 бакалавры академиялық дәрежесін беруге лайық деп есептеймін.

Ғылыми жетекші: «Программалық инженерия» кафедрасының техниккалық ғылымдарының магистрі, лектор _____ Қ. Марғұлан.

«17» 05 2019ж

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

«Программалық инженерия» кафедрасы

5В060200 – «Информатика»

БЕКІТЕМІН
Кафедра меңгерушісі
тех. ғыл. кандидаты,
ассистент-профессор
_____ Р. Юнусов
"20" мая 2019 ж.

Дипломдық жоба орындауға
ТАПСЫРМА

Білім алушы *Мұсабекова Ботагөз Сәбитқызы*
Тақырыбы *«Оқу пәнінің электронды каталогы» веб-қосымшасы*
Университет академиялық мәселелер жөніндегі проректорының бұйрығымен *«16» қазан 2018ж. № 1162-б шешімімен бекітілген.*
Аяқталған жобаны тапсыру мерзімі *2019 жылғы «21» мамыр*
Дипломдық жобаның бастапқы берілістері *Ұсынылатын дипломдық жобада «Оқу пәнінің электронды каталогы» студенттің жеке оқу жоспарын бекітетін веб-қосымшасын жасау.*
Дипломдық жобада қарастырылатын мәселелер тізімі:
а) аналитикалық бөлім
б) жобалау бөлімі;
в) өңдеу құралдары және жобаны талдау;
г) қолданбалы бөлім;
Сызба материалдарының тізімі (міндетті сызбалар дәл көрсетілуі тиіс)
Жобаның презентациялық 31 слайды ұсынылған.
Ұсынылатын негізгі әдебиет *16 әдебиеттер тізімінен.*

Дипломдық жобаны дайындау

КЕСТЕСІ

№	**Дипломдық жобаның орындалатын этаптарының атаулары**	**Ғылыми жетекшіге және кеңесшілерге ұсыну мерзімі**	**Орындалған жөніндегі белгі**
1	Дипломдық жоба тақырыбына байланысты әдебиеттерді іздеу және талдау, дипомдық жобаның негізгі бөлімі ретінде рәсімдеу.	20.01.2019	орындалды
2	Дипломдық жұмысты жобалау: UML қолданушы, тізбек, класс....т,б диаграммаларын дайындау.	10.02.2019	орындалды
3	Дипломдық жобаның веб қосымшасын дайындау, іске асыру және тестілеу.	05.03.2019	орындалды
4	Дипломдық жобаны диплом алды есеп алу комиссия мүшелеріне таныстыру, ескертулерін түзетіп дипломдық жобаны аяқтау. Қорытынды жасау	02.04.2019	орындалды
5	Көрнекі материалдар мен презентацияны дайындау	20.04.2019	орындалды

Дипломдық жоба бөлімдерінің кеңесшілері мен норма бақылаушының аяқталған жобаға қойған

қолтаңбалары

Бөлімдер атауы	**Кеңесшілер Аты, әкесінің аты, тегі (ғылыми дәрежесі, атағы)**	**Қолтаңба қойылған мерзімі**	**Қолы**
Бағдарламалық бөлім	Сман Н. О. Ассистент	16.05.19	
Нормалық бақылаушы	Алғожаева Р. С. Сениор-лектор	20.05.2019	

Ғылыми жетекшісі_____Қ. Марғұлан.

Тапсырманы орындауға алған білім алушы _____ Б.С. Мұсабекова.

Күні «20» қаңтар 2018 ж.

АҢДАТПА

"Оқу пәнінің электронды каталогы" web-қосымшасын құру жобасында – жоғарғы оқу орнындағы білім алушылардың жеке оқу жоспары, шартқа сай міндетті және элективті дисциплиналары таңдау арқылы бекітіледі, сонымен қатар академиялық қарыз курстарына тіркелу ұғымы қарастырылған.

Дипломдық жоба кіріспеден, негізгі үш бөлімнен және қорытындыдан тұрады:

Кіріспеде таңдалған тақырып актуалдылығы қарастырылып, жобаның негізгі мақсаты мен тапсырмасы, зерттеу объектісі анықталады.

Бірінші бөлімде студенттің жеке оқу жоспары, сонымен қатар міндетті және элективті дисциплиналар жайында, осы ұғымдардың мүмкіншіліктері, қолданылуы жайында жазылды, сәйкес веб-қосымшамен өзгешелігі ашық сипатталды.

Екінші бөлімде "Оқу пәнінің электронды каталогы" web-қосымшасына UML диаграммалар келтіріліп, қосымшада қолданылған орталар және бағдарламалау тілдерімен жұмыс істеу негіздері қарастырылған. Жоба бірнеше технологиялар арқылы құрастырылды.

Үшінші бөлімінде веб-қосымшаның автоматтандырылуы және жеке оқу жоспарының интерфейсі, мүмкіншіліктерінің скриншоттары көрсетілген.

"Оқу пәнінің электронды каталогы" түсініктемесі жалпы 43 бет, оның ішінде 24 сурет, 1 қосымшадан тұрады. Жұмысты жазуға 47 әдебиеттермен мақалалар қолданылды.

АННОТАЦИЯ

В проекте "электронный каталог учебной дисциплины" реализуется выбор дисциплин для индивидуального учебного плана состоящих из обязательных и элективных дисциплин в соответствии с договором обучения, а также предусмотрено понятие регистрации на курсы академического долга.

Дипломный проект состоит из введения, трех основных разделов и заключения:

В введении рассмотрена актуальность выбранной темы, определяются основные цели и задачи проекта, объект исследования.

В первой части подробно описаны особенности индивидуального учебного плана студента, а также обязательные и элективные дисциплины, также возможности, применение этих понятий, в соответствии с соответствующим веб-приложением.

Во второй части приведены диаграммы на языке UML к web-приложению "электронный каталог учебной дисциплины", где предусмотрены основы работы с используемыми в приложении средами и языками программирования. Проект был разработан с помощью нескольких технологий.

В третьей части представлена автоматизация web-приложения и интерфейс индивидуального учебного плана, также скриншоты возможностей приложения. В приложении прописаны интерфейсы и пояснения.

Объяснение "электронный каталог учебного предмета" состоит 43 из страниц, из них 24 рисунков, 1 приложение. К написанию работы использовались 47 литературных статей.

ANNOTATION

In this project, the "electronic catalog of the discipline” is approved by selecting an individual curriculum, mandatory and elective subjects in accordance with the contract of training, and also provides for the concept of registration for courses of academic debt.

The diploma project consists of an introduction, three main sections and a conclusion:

In the introduction, the relevance of the chosen topic is considered, the main goals and objectives of the project, the object of study are determined.

The first part describes in detail the features of the individual curriculum of the student, as well as mandatory and elective subjects, as well as the possibility of applying these concepts, in accordance with the relevant web application.

The second part contains diagrams in the UML language to the web-application "electronic catalog of the discipline", which provides the basics of working with the environments and programming languages used in the application. The project was developed with the help of several technologies.

The third part presents the automation of the web application and the interface of the individual curriculum, as well as screenshots of the application features. The Annex stipulates the interfaces and explanations.

Explanation "electronic catalog of the subject" consists of 47 pages, including 24 figures, 1 Appendix. 16 literary articles and articles were used to write the work.

МАЗМҰНЫ

	Кіріспе	8
1	Аналитикалық бөлім	9
1.1	ЖОО-да студенттердің жеке оқу жоспарын тіркеуді автоматтандыру жүйелеріне шолу	9
1.2	Автоматтандырылған жеке оқу жоспарын тіркеуші қосымшаларын салыстыру	10
2	Автоматтандырылған жеке оқу жүйесіне тіркелуді жобалау	15
2.1	Жоба қолданушы интерфейсі мен орындалуын UML-де жобалау	15
2.2	Жоба орындалуына бағдарламаларды дайындау	23
3	Қолданбалы бөлім	29
3.1	Жеке оқу жоспарын тіркеуші веб қосымшаны іске қосу	29
3.2	Күзгі және көктемгі оқу семестріне тіркеуді ұйымдастыру	30
3.3	Академиялық қарыз курстарына тіркелуді ұйымдастыру	33
	Қорытынды	35
	Пайдаланылған әдебиеттер тізімі	36
	А Қосымшасы. Техникалық тапсырма	37
	Б Қосымшасы. Бағдарлама мәтіні	39
	Спецификация	47

КІРІСПЕ

Бүгінгі таңда ақпарттық технологиялар жүйесі жиі жаңарып, үлкен ауқымды жаңа дүниелерді ортамызға қосуда. Жаңа технологиялар адамның күнделікті өміріне қажетті және уақытты тиімді қолдануға мүмкіндік береді. Кредиттік технологиямен оқыту барысында білім алушының жеке оқу жоспары құрастырылады. Жоғарғы оқу орнының қабырғасында білім алушының өзінің қарамағына алатын алғашқы құжаттардың бірі – Жеке Оқу Жоспары.

Жеке оқу жоспары – элективті дисциплиналар каталогы және оқу жоспары негізінде группа жетекшісінің көмегімен, студенттің әрбір жылдық оқу жылына арнайы, өзі толықтай құрастыратын пәндер тізгіні. Білім алушы ЖОЖ бекітуге және оқу жоспарының шарттарына сай берілген пәндерді меңгеруге жауапты. Жоғарғы оқу орнының порталында жеке оқу жоспарын құрудың өз ережелері мен қадамдары болады. Бұл процесс әр оқу жылында орындалады және әр семестрге қойылған кредиттік шектеулер бар, ал сол кредиттік норманы толығымен игермеген жағдайда, қосымша семестр ұғымы қарастырылады. Білім алушының оқу жылындағы академиялық қарыздарынан құтылу үшін, жеделдетілген сондай-ақ, кредит санын толықтыру үшін ақылы түрде өтетін қосымша семестр ұйымдастырылады. Қосымша семестр ұзақтығы 6 аптаға созылады, ескере кететін жағдай, семестр тек ақылы түрде оқытылады. Оқудың ақысы меңгеретін пәннің кредиттік санын негізге алып, оқу жылының бекітілген бағасына байланысты анықталады.

Дипломдық жобаның тақырыбы: Оқу пәнінің электронды каталогы.

Жоба білім алушының жұмыс оқу жоспарын, жеке оқу жоспарын электронды түрде оқу порталында жүзеге асырады. Сонымен қатар, жеке оқу жоспары бөлімінде қосымша семестрге тіркелу операциясы электронды түрде орындалады. Электронды түрде қосымша (жазғы) дисциплинаға тіркелу, білім алушылар мен оқытушыларға, сонымен қатар тіркеуші бөлім қызметкерлеріне өте тиімді технология. Жоба тиімділігімен бірге дизайны да жағымды болу керек. Жобаның web-парақшасына қойылатын талаптар, мынадай болу керек:

- қауіпсіз;
- жағымды дизайнымен;
- қолдануға қолайлы;
- қарапайым;
- кез-келген браузерде ашылатын;
- адаптивті.

1 Аналитикалық бөлім

1.1 ЖОО-да студенттер жеке оқу жоспарын тіркеуді автоматтандыру жүйелеріне шолу

Жеке оқу жоспары – элективті курстар каталогы және оқу жоспары негізінде группа жетекшісінің көмегімен, студенттің әрбір жылдық оқу жылына арнайы, өзі толықтай құрастыратын пәндер тізгіні.

Элективті курстар каталогы мен оқу пәндер жоспары негізінде студент өзінің білімдік траекториясын жүзеге асырады. Студент шартқа сай міндетті турде оқылатын және элективті курстарды таңдап, өзінің жеке оқу жоспарында көрсетеді. Студент жеке оқу жоспарын құруға және оқу жоспарының шарттарына сай берілген пәндерді меңгеруге жауапты.

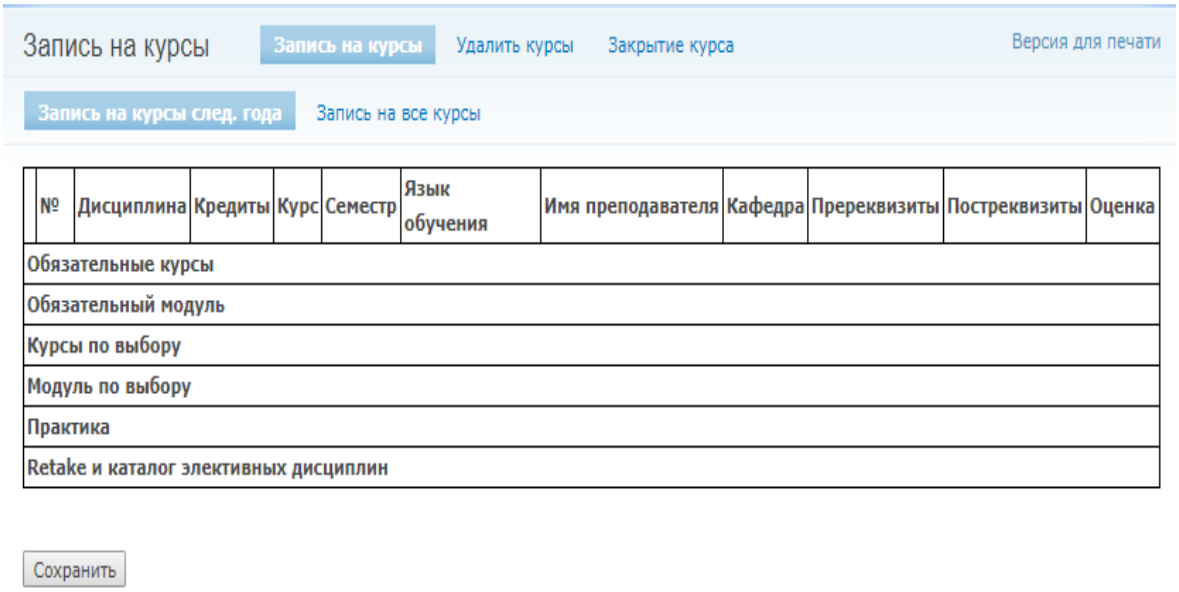
Жеке оқуды жоспарлауды студент эдвайзерімен бірлесе отырып, 1 жылға қалыптастырады. Осы процесс барысында кафедра студентке оның эдвайзері арқылы пәндер тізгіні бойынша, берілген элективті курстар мен олар жайында қысқаша түсініктері туралы, толықтай ақпаратты белгілі уақытында беріп, тіркеу кезіне дейін университет жүйесінде пәндер презентациясымен қамтамасыз етуге міндетті.

Оқу кезеңін бастар алдын, университет жүйесінде, қолданыстағы оқу жоспарына сәйкес, пәндердің толықтай силлабусы енгізілген болуы қажет. Жеке оқу жоспарын құру барысында студентке, мамандығының жұмыс жоспарына ұқсас, элективті курстар мен міндетті пәндер тізгіні беріледі. Оқу пәнін таңдар алдын, міндетті түрде курстардың тізбектестігін есепке алу қажет, себебі семестрде студент пәннің пререквизиттерін игермесе, ол пәнді тіркей алмайды.

Білім алушының жеке оқу жоспарының арасында, өзге мамандықтардың бекіткен базалық блоктың курстары да кездесуі әбден мүмкін. Құрылған жеке оқу жоспары 3 данадан тұрады да, студент қол қойған соң, келісім үшін эдвайзерге тапсырылады. Қателіктер болмаған жағдайда, эдвайзер қол қойып, тіркеуші кенсесінің қолдауымен, деканға ұсынады. Жеке оқу жоспарына қол қойылған соң бекітіп, бір данасы студентте, екіншісі – эдвайзерде сақталады да, студенттің оқу жоспарын жүзеге асырылуын бақылауға негіз ретінде пайдаланады. Жеке оқу жоспарының нақтылай айтқанда, бірінші данасы – кафедрада сақталады да, екіншісі – аралық емтихан құру үшін тіркеу бөліміне жіберіледі, үшіншісі – білім алушыда сақталады.

Жеке оқу жоспарын бекітудің соңғы мерзімі – семестрдің алғашқы аптасы. Студент жеке оқу жоспарына id-нөмірі арқылы тіркеле алады және күнтізбеде берілгендей, жеке оқу жоспарын тіркеуді жүргізу барысында, теориялық оқу кезеңі басталмас бұрын, қайта тіркеу мерзімінде, мамандықтардың жұмыстық оқу жоспардың мөлшерінде өзгертуге мүмкіншілігі бар.

Білім алушының жеке оқу жоспарының ішінде, мамандықтардың жұмыстық оқу жоспарлары мен пәндер кестесі жасалады. Жұмыстық оқу жоспары әр мамандықта 3 бөлімнен тұрады, олар: жалпы, негізгі және де кәсіптік бөлім. Әр бөлімде міндетті түрде таңдалу керек компонентті пәндер тізгіні көрсетілген және кредиттің саны бойынша таңдау компоненттер, оны зерттеу кезегімен, оқу пәндерінің түрлері және бақылау саны берілген. Әр сабақ, негізі заң бойынша бір семестр толығымен оқытылады. Оқу жоспары сабақты жүйелі анықтау ережесі арқылы құралады, сонымен қатар реквизиттер туралы ескере отыру керек. 1.1-суретте ҚазҰТЗУ жоғарғы оқу орнының ЖОЖ бөлімі көрсетілген.



1.1-сурет – Жеке Оқу жоспары бөлімі

1.2 Атоматтандырылған жеке оқу жоспарын тіркеуші қосымшаларын салыстыру

Сәтпаев университетінің жеке оқу жоспары – білім алушының меңгеретін пәндер тізімін таңдап бекітуге мүмкіндік береді. Бұл жоғарғы оқу орнының порталында жеке оқу жоспарын құрудың өз ережелері мен қадамдары бар.

Білім алушының оқу жоспарында дисциплиналардың тағайындалуы, жалпы кредиттік санымен қарастырылады және кредиттің негізгі саны беріледі. Білім алушыға дисциплиналарының қатарына әр жылда оқып жатқан оқу орны, қосымша элективті дисциплиналардың тізімін дайындайды. Элективті дисциплиналардың каталогын дайындауда, элективті дисциплина тізімін жүзеге асырудағы, қызмет атқарушылар білім берушілердің пікірлерін ескереді.
Элективті дисциплинаның тізімін әзірлеу процесі мына пунктерден тұрады:

кез-келген білім беру кафедрасы дайындаған дисциплиналарды береді;

– әр дисциплинаға пререквизит пен постреквизит және ұсынылған дисциплинаға қысқаша түсінікті бірге бекітеді.

Білім беру бағдарламасының жасалуы үшін, жауапты шығарушы кафедра, бұл мамандық бойынша ұсынылған барлық пәндерді зерттейді және кафедра мәжілісінде, жұмыс берушілер мен жоғарғы курстардағы білім алушылардың қатысуымен, өз саласында белсенді өмірлік позицияны танытатын кәсіпқойдың қалыптасуына әсер тигізетін пәндерді элективті пәндердің каталогына енгізу туралы шешім қабылдайды. Бұл элективті дисциплиналардың тізіміне бекіту жайындағы үкімді кафедра басқарушысы бекітеді.

Элективті дисциплиналардың бекітілген өзінің кредиттік мөлшері қарастырылған дисциплинаға байланысты, мәлімет мөлшеріне, оның қиындығына қарай болады.

Элективті пәндердің каталогын оқу орнының порталы, яғни білім алушының жеке парақшасымен қарастыруға мүмкіндігі болады, оның көшірмелері тіркеуші бөлімшелерінде болады. Мамандықтың элективті пәндер каталогы ыңғайлы, жан-жақты прогаммаларды дайындауға бағытталған. Студенттерге қарастырылған, толық оқыту жолын жүзеге асыруда квалифиционды көмек беруге жетекшілердің жұмысы ескерілген.

Элективті дисциплинаны меңгеру реті қарапайым дисциплинаны меңгеруге ұқсас, сонымен қатар бұл ұғымда да пререквизиттер мен постреквизиттер ескеріледі.

Элективті пәндер каталогы қызмет атқару қадамдарын меңгергеннен кейін, студентке берілетін жолдың кіші түсінігі, оқылатын сабақтарды меңгергеннен кейін, жұмысқа тұрудың түрлері бейнеленген.

Элективті пәндер каталогында дисциплиналарды меңгеруге керекті кредит көлемі беріледі. Элективті дисциплинада, бір дисциплина ең аз дегенде үш кредит санынан тұруы керек немесе екі кредиттік пәнді бекітуге болады.

Студенттің таңдауды қарастырар барысында, оған ыңғайлы болу мақсатында пререквизит пен постреквизит берілген. Пререквизит – осы дисциплинаны меңгерердегі дисциплина, постреквизит – осы дисциплинаны меңгергеннен кейін қарастыру керек дисциплиналар.

Элективті дисциплиналлардың тізімі басқа элективті дисциплиналарды таңдауды қарастыруға мүмкіншілік етеді.

Білім алушының оқу жылындағы академиялық қарыздарынан құтылу үшін, жеделдетілген сондай-ақ, кредит санын толықтыру үшін ақылы түрде өтетін қосымша семестр ұйымдастырылады. Бұл қосымша семестр ұзақтығы 6 апта, ескере кететін жағдай, бұл семестр тек ақылы түрде оқылады. Оқудың ақысы меңгеретін пәннің кредиттік санын негізге алып, оқу жылының бекітілген бағасына байланысты анықталады.

Ал бағасы қанағаттанарлықсыз бекітілген пәннің бағасын туралау мақсатында, қайта оқу үшін студент келесі оқу семестр кезеңіне немесе қосымша семестрде сол дисциплина бойынша жұмыс оқу жоспарында бар барлық дисциплина түріне қайта жазылады. Жазылған дисциплинаны қайта игеруге

рұқсат алған жағдайда, қайта таңдалған пәнді игеріп, қорытынды бақылау жұмыстарын тапсырады. Міндетті түрде студенттің қосымша семестрден өтуі жоспарлы түрде жасалады.

Егер білім алушы ақылы түрде пәндерді жазғы семестрде аяқтауды қарастыратын болса, ол таңдайтын пәндердің саны максимум 12-18 кредиттен аспау керек.

Қосымша семестрге рұқсат алу үшін, студент аттестациялау аяқталғаннан кейін, деканнан қосымша семестрге жазылатыны жайында, берілген үлгідегі қағазды толтырады. Толытырып болған соң, жазылған факультеттің деканымен ақылдасады.

Студент қосымша семестрдің төлену керек төлем ақысы, университеттің ынтымақтастық туралы келісімшарты бар банкке төлейді. Бұл міндеттерді растайтын түбіртектің түп нұсқасы мен бланкті алып, университеттің бухгалтерлік бөлімшесіне барады. Бухгалтер бланкке қол қояды. Қол қойылғаннан кейін 3 жұмыс күн аралығында диспетчер бөлімшесіне өткізіледі.

Студенттің сол пәнді оқытатын мұғалімді таңдауға мүмкіншілігі бар. Егер студенттің меңгеретін пәнінде төленген ақысында қарыздары болса, онда ол студентке мұғалімнің сабақ жүргізуіне жол берілмейді.

Жүргізіліп жатқан сабақ бойынша қорытынды бақылауды, қосымша семестр соңында қалыпты оқылатын сабақтар сияқты курсты жүргізген мұғалім алады.

Аралық бақылау, семестрлік қорытынды барлық бағалар есептелген соң, қосымшаның соңғы 3 күнінде емтихан өткізіледі. Ал қосымша семестрде оқылған пәндердің емтихандарын тек бір рет тапсыруға рұқсат. Пәнді оқып аяқтағаннан кейін, қорытынды оқу жылының GPA бағасы есептелінеді.

GPA саны нормадан төмен, қосымша семестр нәтижелері қанағаттанарлықсыз болса, студентке курсты қайта оқуын міндеттейді.

Ал жазғы семестр қорытындысы негізінде GPA балы жетпей, қанағаттанарлықсыз баға алған студент, мемлекеттік грантынан айырылып қалады. Қосымша семестр біткен соң, деканат студенттердің GPA балын растап, курстан курсқа өту жайында бұйрық жазады.

Жеке оқу жоспарын құру әр оқу жылы бір-біріне ұқсас болады. Бірақ ескеретін жағдай, алғашқы оқу жылы, яғни 1-курс студенттеріне алдымен жұмыс оқу жоспарын бекітеді.

Жұмыс оқу жоспарында студенттің меңгеретін пәндер тізімін, топтың жұмыс оқу жоспарын құрады. Жұмыс оқу жоспары 1 ғана рет толтырылады, және өзгертілмейді. Келесі қадамда, студент тіреуші бөліміне барып, оқу жылы басталмай, алдын ала оқитын 1 жылдық пәндер тізімін, яғни жеке оқу жоспарын алады. Жетекшісімен кеңесіп, сол бір аптаның мерзімінде таңдалған пәндер бойынша бекітіледі. 1.2-суретте Жеке оқу жоспарының құрылымы көрсетілген.

Осылай білім алушының 1 жылдық жеке оқу жоспары құрылады. ЖОЖ-ны бекіткен соң, оған өзгерістер енгізу мүмкіншілігі болмайды.

1.2-сурет – Жеке оқу жоспарының құрылымы

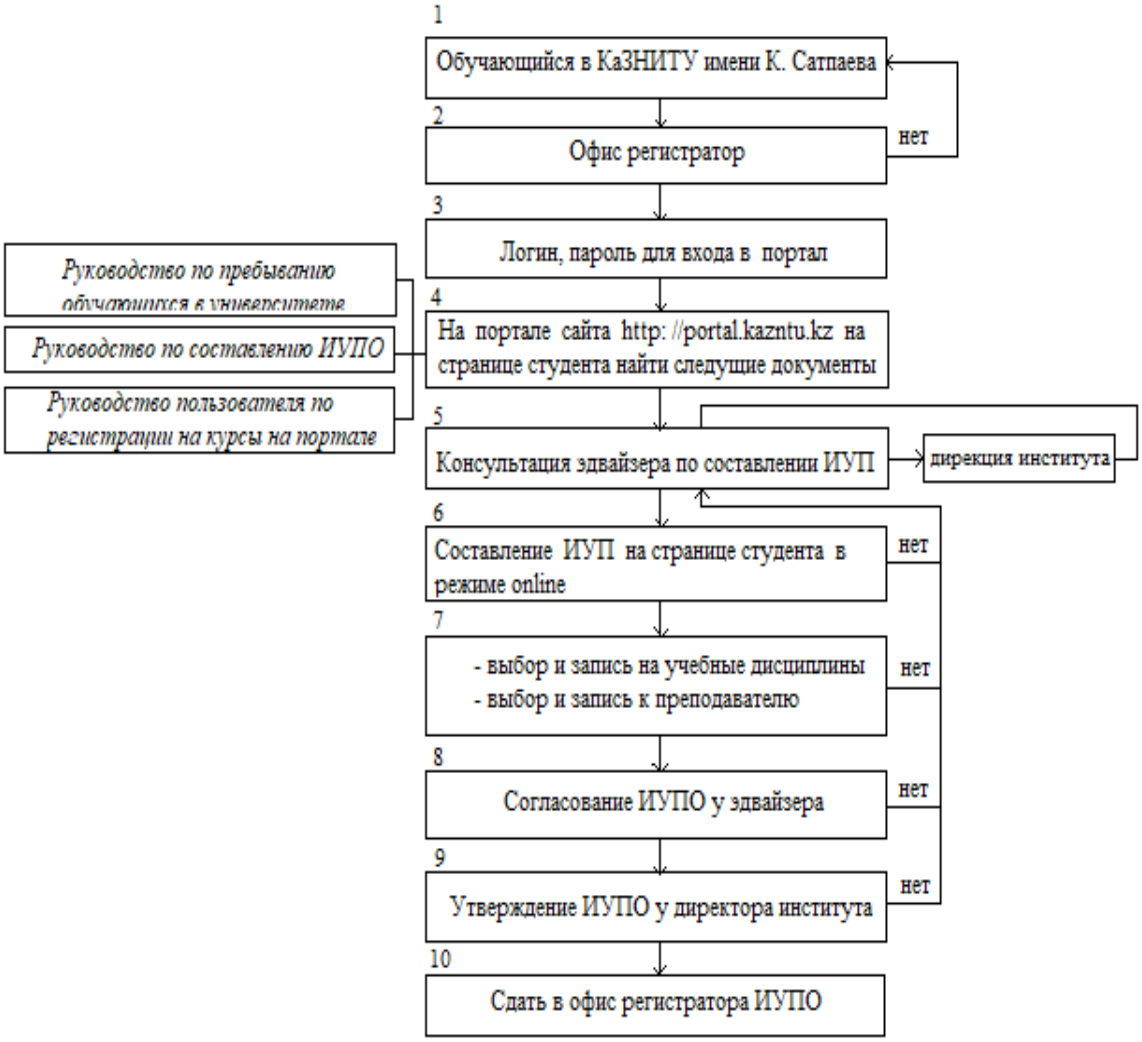
Ал екінші курстан жоғары курс студенттері одан кейінгі оқу жылдарына өзгерту құқықтары бар, және ол үшін берілген уақытқа дейін жазбаша түрде өтініш жазып, басқа пәнге ұсыныс толтырып, өзгертулеріне болады.

Оқу ережесі бойынша, ректорат тіркеуші бөлімшесімен бірлесе келісіп, пәнді оқу үшін керекті білім алушылардың минималды санын және мұғалім үшін академиялық топтың максимумын жазады.

Жиырмасыншы маусымға дейін белгілі таңдалған дисциплинаға студенттер саны жеткіліксіз болса, онда ол пән ашылмайды, ЖОЖ-на бекітілмейді. Тіркеу бөлімшесі бұл хабарламаны университеттің порталының негізгі жариялау бетінде хабарлайды. Ал осы дисциплинаны таңдаған студенттер 25-тамызға дейін тіркеу бөліміне барып, білім алушының ЖОЖ-на өзгеріс енгізуіне ұсынысын білдіреді.

Білім алушы егер өзінің келесі жылға арналған ЖОЖ-ын тапсырмаса, тіркеу бөлімінің қалауы бойынша қабылданатын болады. Барлық семестр кезеңінде өзгертуге болатын қателіктер мен түзетулерді білім алушының өтініші бойынша тіркеу бөлімінде жүзеге асырылады.

1.3-суретте білім алушылардың жеке оқу жоспары құрылу жайында тіркеуші бөлімнің қызметкері мен білім алушының атқаратын қадамдары бейнеленген. Солай-ақ тіркеу бөліміндегі қызметкерлер де, алғашқы қадамдарында өздерінің жеке парақшаларына кіруден бастайды. Дисциплиналарды таңдап, бекіту онлайн түрде орындалады және студент керекті дисциплиналар тізімін, сонымен қатар оқытушыларды таңдайды. Келесі қадамда, студент жетекшісі құрылған тізімін растап, институт директорының келесімін алып, тіркеу бөліміне тапсырады.



1.3-суретте – білім алушының ЖОЖ қалыптастыру реті берілген

2 Автоматтандырылған жеке оқу жүйесіне тіркелуді жобалау

2.1 Жоба қолданушы интерфейсі мен орындалуын UML-де жобалау

Rational rose – ақпараттық жүйелерді дайындау барасысындағы, визуалды модельдеудің ең танымал құралдарының бірі болып есептеледі. Бұл кез-келген класстағы қосымшаны модельдеуге арналған ыңғайлы құрал.

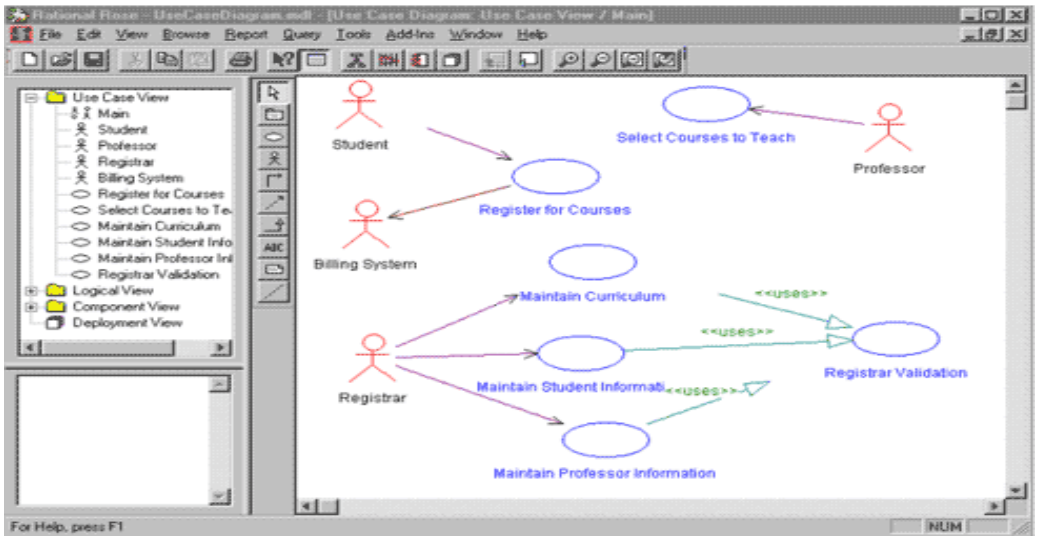
IBM Rational Rose керекті модельді дайындағанынан бөлек, бірге саналатын компоненттердің, сонымен қатар сол компоненттер ортасындағы қосылымдардың дәлдігін, анализ жасауды жүзеге асыруға мүмкіндік береді.

Rational Rose процессті оңайлату және аз ғана уақытта жақсы деңгейдегі бағдарламалық қамтамасыздандыруды тағайындау үшін, модельдеу құралын ұсынады.

Қолданбаны тездету – қосымшаларды әзірлеуді жеңілдетуге мүмкіндік беретін визуалды модельдеу ортасын ұсынады.

Өтінімдерді жобалау мен әзірлеудің интеграциясы – UML моделінің орындалуының және жазылуының ортақ құралымен жобалық топтың жұмысын біріктіреді. 2.1-суретте UML – диаграмма интерфейсі көрсетілген.

Қолданбаны жылдамдату – қосымшаларды тағайындауды оңайлатуға мүмкіншілік тудыратын, осы визуальды модельдеу тілін ұсынады. 2.1-суретте UML интерфейсі суреттелген.



2.1-сурет – IBM Rational Rose интерфесі

Дипломдық жұмысты бастамас алдын, тапсырмаға байланысты жобаның қолданыс аумағын растау керек. Осы себептермен UML тілінде жобаны сипаттаудан бастадым. Жобалау түрлі бағдарламалар мен үрдістерді және оның қатарында объектіге бағытталған бағдарламаларды жобалау модельдерін сипаттауға қажет.

Әр модельдеудің принциптері әр түрлі, прецендеттер модельдеу түрінің басты мақсаты бекітумен өзара байланыста. Принциптер принцип сипаты ішінде мәтіндік түрде сипатталған.

Әсте, баптаудың өмірлік циклі ағымды пайдаланушының принципіне жаңарту енгізілсе, осы жаңартулар принцип сипаттамасына берілгендей, прецендентер моделіне де беру керек.

Кез-келген прецендент оқиғасының мәліметі, тіркелген ағын арқасында мінездемелеу қажет. Ұқсас мәтіндік құжат actor прецендентi бастаған уақытта жүйе қандай қызмет атқаратыны жайында қарастырады. Прецендентiн структурасы әр түрлі, алайда басымшылығында мынандай бөлімшелерінің табылуы міндетті:

- шағын мәлімдеме;
- оқиға жайлы егжей-тегжейлі мәлімдеме;
- басты және баламалы ағындар.

Пайдалану түрін сәйкестендірудің негізі болып, сыртқы мәлімдемелер қызмет жасайды. Бірінші жүйе әсер көрсететін, айналада орын алып жатқан жағдайлардан бастап қарастырған жөн. Кейбір белгілі мәлімдемелер пайдаланушыны кірістірмей, системаның әсерін пайда болуы немесе пайдаланушының өз әсерін қалыптастыруы мүмкін.

Әсер етуді керек қылатын мәлімдемелерді сәйкестендіріп пайдалану түрін растауға ықпалын тигізеді. Тағайындаушы және пайдалану түрлері арасынағы байланыс оған қоса, пайдалану және ұлғайту деген байланыс 2 типті болады. Пайдалану түрінің бірі екіншісіне жақын, алайда одан жеткілікті мөлшерде жүктеме алған кезде ұлғайту типті байланыс пайдаланылады.

Ұлғайту және пайдалану бірлестігінің арасындағы өзгешеліктер. Осы байланыстар бірнеше пайдалану түрлерінің ортақ ережелер фрагменттерін бөлек нұсқаға алып, өзге нұсқауларға пайдаланады, болмаса кеңейтеді. Келесі жағынан, осы күйлер оны өзге түрлі принциптерде орындайды.

Системаны жасауға зат аймағынан хабардар болу керек. Сол сияқты системаның қандай қызмет атқаратыны жайында білу қажет. Диаграмма системасы атқаратын мәміле тізгінін жасауға тағайындалған.

Бұл диаграмма операциясы бекітілген функцияның система компоненті табылған тізгіндер системасын жасайды. Осы система сияқты жүзеге асырылатын системаның функция системасы жасалады, жүйемен байланыстағы компоненттер қызметінің жоспарлары бекітіледі. Прецендент әрекетін сипаттауға, пайдаланушы қызметін тұтас жеткізуіне ориентацияланған.

Прецендентер диаграммасы – система атқаратын қызметінің тұжырымдамасының нақтыланған түрі. Кез-келген прецендент нақты түрде бекітілген мағлұматтар ағымының арқасында модельделу қажет. Сол сияқты оған ұқсас actor прецендентi істегенде системаның қандай қызмет атқару керек екендігін нақтылайды.

Use Case диаграммасы – ол жанданған субъектінің көп саны схемалық көрсеткіші пайдаланудың, осы әйтпесе өзге опцийлардың түрлерімен жалпы қолданылуы. Системаны әзірлегенде жалпы осы системаның негізгі қасиетін

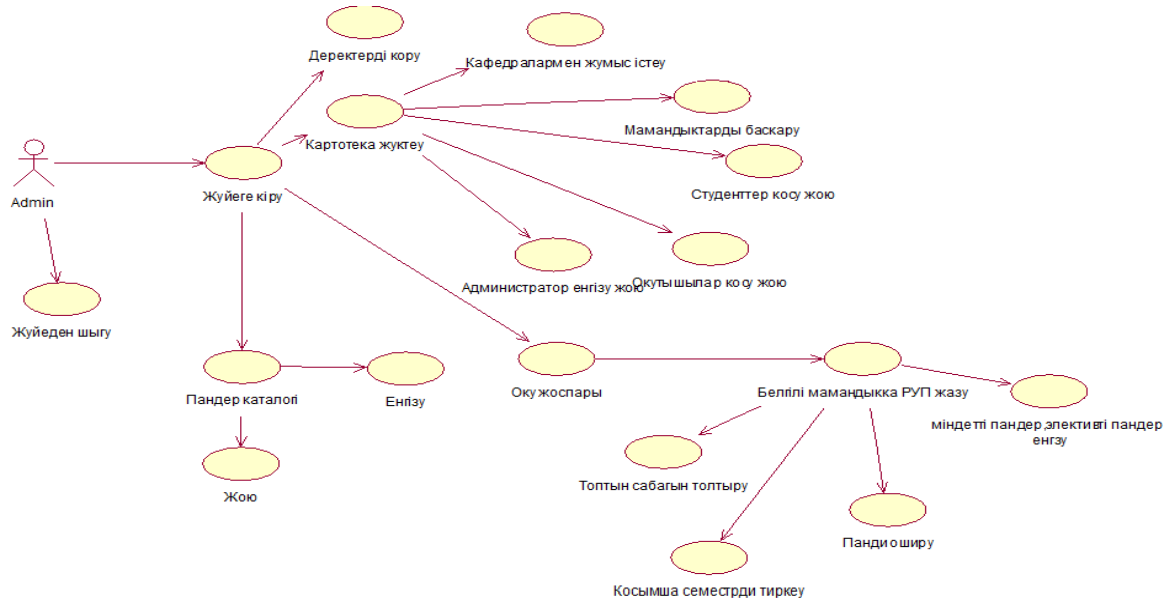
және пайдаланушылардың үлкен саны тапсыратын түпкі диаграмма құралады.

Осы диаграмма нақты actor-ға тағайындалған, нақты бір қорытынды бекітетін, системамен жасалатын жұмыстар тізгінінің моделі. Прецендент жалпы сипаттамадағы маңыздылықтардың ережелік структурасын құрайды және преценденттер бірлесуімен жіберіледі.

Схемалық тұрғыда бұл линиялы бекітілген эллипс жүзінде жіберіледі және осы орынға атауы енгізіледі. 2.2-суретте клиент-серверлік үлгісі көрсетілген

Actor – ол системадағы маңызды компоненттерімен бір байланыстағы компоненттің атқаратын қызметі болып саналады. Actor мен қолданушының өзгешелігі мынандай: қолданушы системаны қолданатын нақты компонент. Actor саны көп болғандықтан, оған көптеген маңыздылығы әсер етуі ықтимал.

Білім беруші алған қорытындыларды Microsoft EXCEL немесе текстілік жазбаға бекітеді. Себебі алғашқы ақпараттарды бақылау – осы дисциплиналар жайында ақпараттар, соларға негізделетін жүктеулер немесе жіберу моделі.



2.2-сурет – Преценденттер диаграммасы

Класс диаграммасы – системадағы класстың атқаратын қызметінің негізін және класстардың өзара арақатынасын сипаттайды.

Класстар диаграммасы – компонентке ориентацияланған позициядағы басты диаграмма ретінде есептеледі. Бұл диаграммалардың негізгі қызметі – системадағы компоненттердің негізін табу және класстардың өзара қатынастарын анықтау. Осы класстар ара-қатынастарының 2 байланыс түрі пайдаланылады: ассосациялы және подтипті.

Байланыстардан түрлерінен бөлек класс диаграммасының құрамына компоненттер, операциялар, нысандар саналады. Класс диаграммасын модельдеуден алдын, осы диаграмманың принциптерін пайдаланатынын растап алу қажет.

Класс диаграммасы деп бірнеше жобалардың көрсетілімі суреттелген

диаграммаларды айтады.

Класстардың маңызды қасиеттері бірыңғай және мультифункционалды. Осы қаситеттердің арасындағы ауқымды мөлшердегі принциптер қызметтің жәй атқарылуына себеп болады.

Класс диаграммалары – бағдарлама кодтарын бекітуді маңызды диаграмма класы арқасында системаның түпкі системасы беріледі. Кейін мәлімет кодтары мен компоненттері айқындалады. Диаграмма классы системаның жалпы интерфейсін модельдейді, нақты система интерфейсі. Класстың нәтижесінде әр кезде диаграмма жаңартылуы әбден мүмкін.

Диаграммалар мен класстардың жаңартулары өздігінен енгізіледі. 2.3-суретте класстардың орны көрсетілген.

Диаграмманың классы әзір система анализдеу сияқты пайдалануы ықпал және дайындауда да. Өзгешеленген кластарда алуан түрлі тәсілдер және жолдар пайдаланылады. Class discovery төрт басты жолдарды нақты көреді. Мынандай жолдардан құралады:

- атауы пайдаланылатын топ маңызды жол;
- классқа жалпы модель пайдаланудағы жол;
- преценденттермен пайдаланудағы жалпы жол;
- жұмыскерлер ерекшелігі, класс, қажеттілік структурасы.

Классқа тиесілі аттрибуттарын модельдеуге байланысты жасалатын системаны жеңіл орнатумен өзгешеленеді.

Класстың жалпыға бір ережелері мен функциялары және одан бөлек нысандармен өзара ортақ байланысы орныққан нысандар құрамын қарастырады. Класстың нысанды жасауға тағайындалған үлгісінде бақылауға болады.

Бұл диаграмма таңбалары интерфейс көрінісінің жалпы өзара баланыстылығын күрделі мұрагерлік системада суреттеуге мүмкіншілік етеді. Диаграммада енгізілген типі система компоненттері сипатталатын кооперативтілік диаграммаға қайшы келеді. IBM RR диаграммасының енгізілген типімен сан алуан тәлімдеме орнатуға болады:

- класстар ортасындағы байланыс жүргізу;
- бірігу байланыстары мен мұрагерлік байланыстар;
- бағдарланған, мерзімдік, стандартты қатынастар;
- класстарды нақтылайды.

Классты негіздеуге алдымен оны нақтылау қажет. Модельдеу класстардың структурасынан бөлек мағынасын толық ашып жазу қажет. Класстық диаграммада класс, көрініс, элемент, мультиоперациялар мен өзара ішіндегі байланыстар расталады. Объектіге-ориентацияланған системаны сипаттау барысында осы диаграмма жиі пайдаланылады. Класстық диаграммалар сипаттау тарапынан системаның стационарлы типіне ұқсас болады.

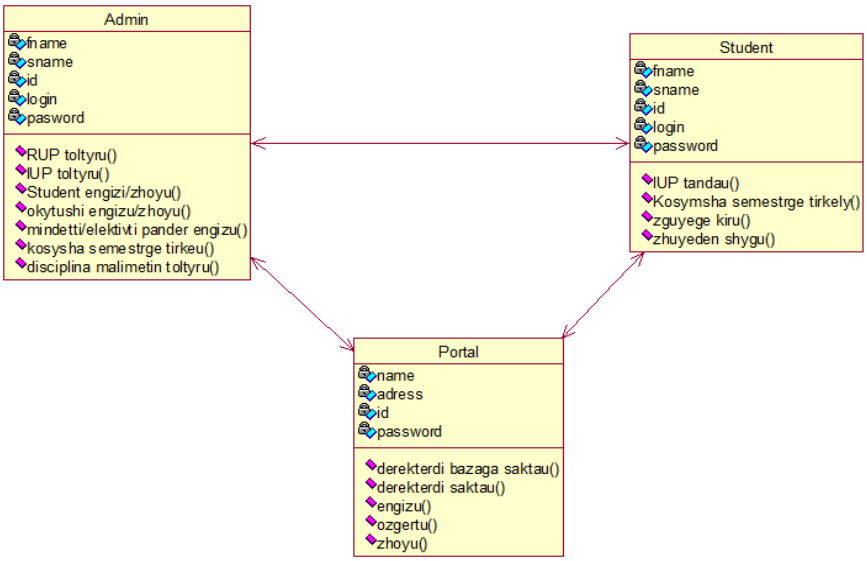
Класстар диаграммасы объектілі-бағытталған модельдеуде маңызды элементі болып есептеледі. Класстар диаграммаларда жақтаулардың ішінде бейнеленеді, ал ол жақтаудың құрамы 3 бөліктен тұрады.

Жоғарғы бөлігінде класс атауы жазылады. Класс атауы жақтаудың ортасында бекітіліп, қарапайым шрифтпен жазылады. Класс абстрактілі болған жағдайда оның атауы курсивпен бекітіледі.

Ортанғы бөлігінде класс атрибуттары, қандай атрибуттар осы классқа тиесілі, типтерімен бекітіледі. Класс атрибуттары кіші әріппен беріледі.

Диаграмманың соңғы бөлігінде класстың қандай әдістері бар бекітіледі. Класс атрибуттары секілді, класс әдістері де кіші әріптен басталады.

Класстар диаграммасы әрекеттестік тарапынан системаның стационарлы типіне ұқсастырылады, яғни активті класстан құралған диаграммалар.



2.3-сурет – Класстар диаграммасы

Нысандардың, яғни объектілердің байланысын белгілі мерзім арасындағы бақылайтын, және нысандар ортасында мәлімдемелерді жіберу мен қабылдап алуға тізбек диаграммасы пайдаланылады. Бұл диаграммаларды нақтылау үшін, алғашында нысандарды нақталау қажет. Пайдаланудағы диаграмманы растаған уақытта, система арасында қызмет атқаратын көріністерді нақтылағанмын.

Осы әрекеттерді тізбектер диаграммаларында атқаратын көріністер actor-лер түрінде енгізіледі. Бұл әрекеттер физикалық көріністер ретінде қарастырылады.

Тізбектердің диаграммаларында нақты байланысқа пайдаланылатын нысандар нақтыланады. Нысандар - ол дайындау арасындағы пайдаланылатын класстар ретінде есептеледі. Осы кластарға ұқсас нысандарды нақтылауға болады:

– мәлімдемелерді басқару системасы;

– системамен серверді жалпылау системасы;

системамен қолданушыны жалпылау системасы.

Кез-келген расталған нысандар, тек өздерінің пайдалану түрлерінде пайдаланылады. Нысандардың арасында бірігу операцияларының жүзеге асуы тізбекті диаграмманың қадамын ресімдеу және жоспарлармен қассиеттердің жүзеге асу барысындағы өзара нысандардың ауысатын жолдаулар тізбегінің сортталған типінде сценарий ретінде саналатын нысандар мен кластар сипатталады.

Осы диаграммалар жалпы жобаның мақсатын ашық түсіндіру топтамасындағы арналған пайдалану түрлерінің іске асуымен бірігуі. Тізбек диаграммаларанның 4 компоненттерден құралады:

– преценденттегі реттік текстің қимылы. Кәдімгі мәтін ретімен, сол жақтан бастап жазылып бастайды. Осы қимыл модельдеуі ретінде қызмет бараысындағы жүзеге асырылатын мәліметтер жұмыс атқарады;

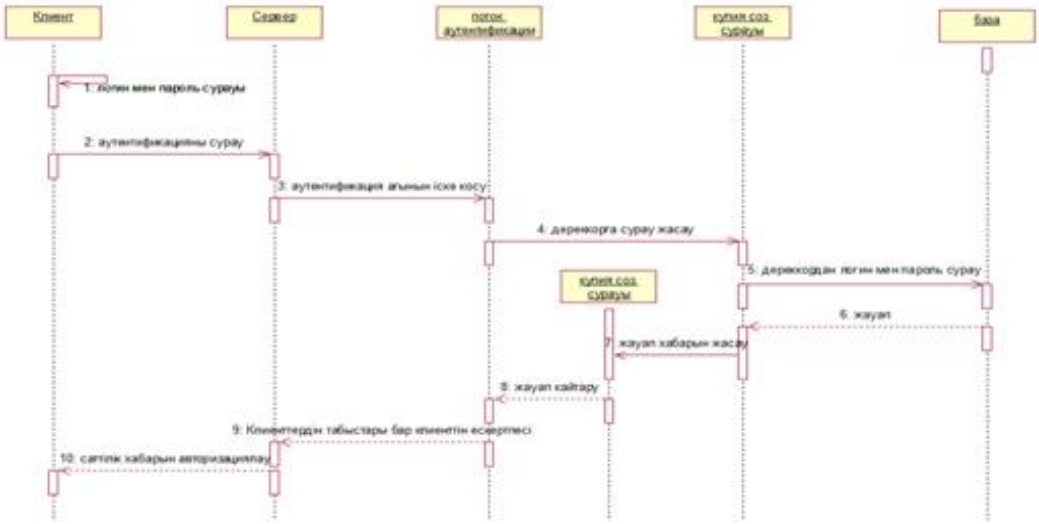
– нысандар «объект-класс» түрінде атуы не нысан санының нөмірі мен класс нысанының атауы бекітіледі;

– белгілі бағытталған нысаннан келесідегі ориентацияланған қимыл жайында мәлімет хабарлауынан тұрады. Арнайы мерзімде іске қосылатын және сол қимылдағы системаның негізгі әсері де болады;

– Операциялары. Тік бұрыш түрінде берілген.

UML диаграммаларының принциптерімен ұқсас нысан тізбек диаграммаларында тік төртбұрыш типінде берілген. Төбеде берілгендей нысанды үш методпен айтуға болады: атын белгілеу, нысан мен кластың атын белгілеу немесе кластың атымен белгілі өлшем керек.

Негізі тізбектер диаграммасы қаншалықты жеңіл сызылса, соғұрлым түсіну де жеңіл болады. 2.4-суретте ЖОЖ құрылу үлгісі сызылған.



2.4-сурет – ЖОЖ құрылу үлгісі

Олар объектілерді, хабарламарды және олардың алмасуын, сценарийлермен, функцияларды бірден көруге рұқсат етеді.

Тізбек диаграммалары нысандарды, мәлімдемелерді сонымен қатар өзара алмасуларын, қаситеттерді көруге мүмкіншілік береді.

Тізбек диаграммасы әрекеттердің рет-ретімен бейнелену әдісі, нысандардың бірігуімен көрсетілгендей, тізбек жолдамалардың нысандар қасиеттерінің іске қосылу кезінде орын ауысады.

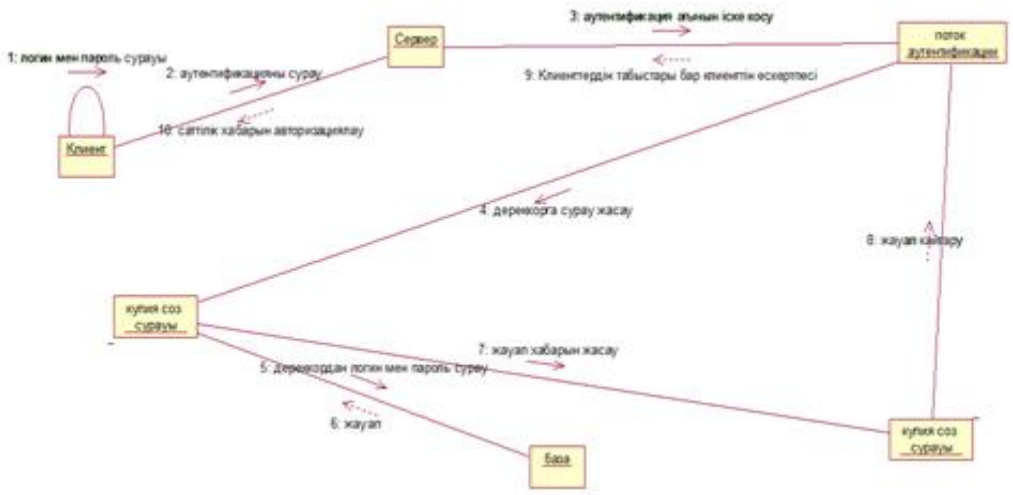
Диаграмманы сызуда объекттер төртбұрыш болып сызылады, сол объектілердің арасында өмірлік жолы яғни орындалатн әрекеттері жазлады. Бұл диаграмманың компоненттері белгілі бір мезеттің бөлшегі, объекттің қарқынды, я болмаса нақты бір қаситетті бейнелейтін тіктөртбұрышты нақтылайтын обьекттердің жолы. Нысандар ішіндегі белгілер мен мәлімдеме ауысуды бейнелейтін көрсетулер.

Кооперация диаграммасы – бұл хабарламаларды жіберетін және қабылдайтын объекттерді құрылымдық жақтан жан–жақтылы ұйымдасуын айқындайтын өзара әрекеттесу диаграммасы. Кооперация диаграммалары жүйе жұмысы барысында объектілердің өзара әрекеттесуін бейнелейді. Мұндай диаграммалар жүйе тәртібінің сценарийлерін модельдейді. Объект атының асты сызылады және әрдайым беріледі, ал қасиеттері таңдаулы түрде көрсетіледі.

Осындай диаграммалар система ережесінің принцибін мәлімдеп, сол сияқты обьектілерді қабылдап құрылымдық әрекеттесуін анықтап әрекеттесу диаграммасы. Осы диаграмма – жүйенің қызмет барысында обьекттердің әрекеттесуін бейнелейді.

Кооператив диаграммалар кез-келген түрлі нысандарға тарихтарда әсерін тигізіп, өзара байланыстық диаграммаларын құрайды.

Кооперативтік диаграмманы есептеуге, жалпы ереже нақтылауына пайдаланылады, яғни солардың ортасындағы ара қатынасты растап, бағдарлама пайдаланушыларды растайды. Түпкі, сонымен қатар сипаттамалық мәлімдемелерін жібереді, нақтырақ класс диаграммасын жүктеуге негізгі жалпы мәлімдемелерді санауға болады. Мультиоперация негізделуіне ықималдылық етсе, диаграммалар компоненттердің сарапталуы мен ара-қатынысын бейнелейді (2.5-суретте көрсетілген).



2.5-сурет – Кооперация диаграммасы

Кооперация диаграммасына қажет функциялары расталады, обьектілер ішінен мультиоперативтік қосылуы бейнелеген. Кооперация диаграммаларды нысандардан тұрады.

Сол үшін ссылкалар жолдаманың қорытындыларымен бірігеді және ссылкалар ретінде саналады. Осы сатыда обьектілер ғана беріледі, сондай-ақ процессті болмаса сипаттаушыны жүзеге асырумен байланыста болады.

Күй диаграммасы – жалпы диаграммалардың сипатын көрсететін әр әрекеттерін бейнелейтін диаграмма. Біріктірілген қимылы мен әр жұмыстар атқарылған компонеттер жүзінде сипатты растау түрінде анықталады. Жеке топ өзге қабылдауларға жіберу әрекетін өздігімен байланысты. Қызмет диаграммалар модельдеу бизнес-процестерді, процесс, параллель және тізбекті есептеу үшін пайдаланылады. Бұл диаграммалар сипаттау операцияларын тізбекті санауға қолданылады.

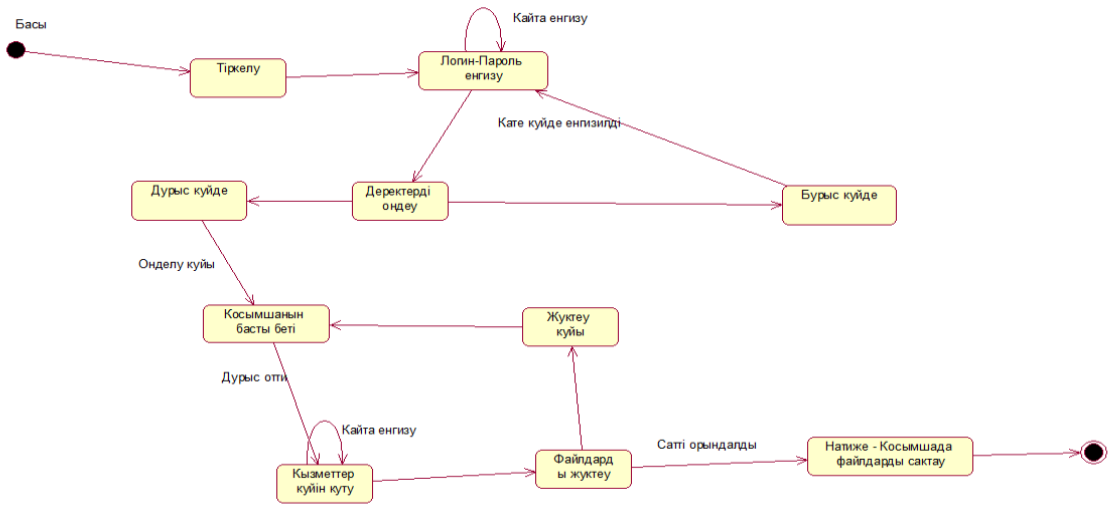
Күй диаграммасы бағыттаушы фигураларын қосылып бөлшек саны мелгілі мөлшерден құрылады. Нәтижесі:

– тіктөртбұрыш, атқарылатын қызметі;
– қабылдаудар;
– ауқымды бірнеше линиялы бөлшектенген;
– қара дөңгелек операциясының бастамасы;
– нәтижелі шеңбер.

Нәтижелері операциясының алғашынан соңына шейін жүзеге асуы мен маңызды жолдарын я болмаса жолдарын бейнелейді (2.6-суретте берілген).

Осы диаграмма – кез-келген нысандарды қалыптастыру барысында нақты іс-қимыл, я болмаса өзге процесстің орындалуын кутүдегі қоныстандыруды іске-қосатын топтама принципі.

Activity диаграммасы – нысанның нақты күйден өзге күйіне алмасуын арандалатын, сонымен қатар қызметтері күйін ауысуымен ескертілген, процесстердің, жолдамалардың нысандық күйлерінің сызықтық сұранысы.



2.6-сурет – Күй диаграммасы

2.2 Жоба орындалуына бағдарламаларды дайындау

Денвер – қазіргі уақыттағы танымал локальдық серверлердің бірі және бұл локальдық сервердің беретін мүмкіншілігі сайт жасағанда желіге қосылуды қажет етпей етпейді.

Денверді жазу уақытында оның 3 батырмасын жұмыстық бетіне орнатып коямыз. Локальды серверді жүзеге асыру үшін старт денвер-ді басып қосамыз. Содан соң жұмысымызды бітіргеннен кейін стоп-денвер батырмасын басып жобамен жұмысты аяқтаймыз. Ал егер біздің енгізген жаңартуларымыз файлды қосқан уакытта денвер таппаса рестарт-денвер батырмасын басамыз.

Батырмалар жұмыс бетінде болмаған жағдайда, С дискіде веб-серверс файлының арасынан денвер файлын басып, денверді іске қосу үшін Run батырмасын, ал жабу үшін Stop батырмасын басамыз. Restart батырмасын айтып өткендей жаңартулар енгізгенде басамыз. Локальды серверді ашқанда 2 қара графикалық фигура көрініп қайта өшіп кетеді. Осы уақытта астынғы бөлікте сағат болатын жерде 2 – денвер-шляпасымен қарындаш шығады. Осы локальды сервердің жүзеге асуын білдіреді.

Денвердің базалық пакеті тұрады:

– инсталяторлар;
– SMTP-сервер және send-mail эмулятордан, PHP, Parser-мен бірге, т.б.;
– Apache веб-сервері;
– MySQL басқару жүйесі phpMyAdmin веб-интерфейсі арқылы;
– MySQL 5 транзакция арқылы;
– Қосу жүйесі мен аяқтаушы денвер компонеттерімен;

PHP (Personal Home Page) – дүние жүзілік қолданыстағы танымал бағдарламалау тілі. Бұл бағдарламалық тіл HTML құжатының өзіне немесе өз алдына бөлек жазуға рұқсат етілген және деректер қорымен жұмыс жасайды. Браузерден бөлек серверде жүзеге асады және синтаксисі жеңіл. Қазіргі тағда мыңдған программисттер қолданатын бағдарламалық тіл.

PHP әрдайым өзгеріп отырады, сондықтанда қазіргі таңда жоғары дәрежелі программалық тілдердің қатарынан қалыспайды. LAMP – web-парақшасын жасауға арналған тізгінге ApacЋE, PHP, MySQL, жатады.

Браузерге тағайындалған программалау аймағында PHP – танымал скрипттік тіл болып саналады, бұл тіл жеңіл әрі тез жоғары дәрежедегі қасиеттеріне, сонымен қатар осы тілдің лицензясының жалпы тармақталатын алғашқы жобаларына қарай, PHP түйінінің қатысуымен, бірегетін сипаттаулардың қатысуымен, ұлғайтулардың қатысуымен ерекшеленеді, мәліметтер қорларымен, PDF форматындағы файлдармен және т.б қызмет атқаруға тағайындалған.

Әр адамға оның тек өзіне тиесілі ұлғайтуын жасап, толықтыра алады. Мыңдаған ұлғайтулар бар, алайда қалыпты топқа жалы жиырма шақты көп жіберілген ұлғайтулар жатады.

PHP уебсерверге немесе соған тиесілі модульдың көмегімен бірігеді. Тағы

өзге АмигоОС, МакОС Х, Юникс, Линукс системаларында шығаруларды шығаруға пайдаланады. Осы уақытта PHP-мен мыңдаған бағдарламашылар пайдаланады. Структура web жобалары ортасында web бағдалмаларды дайындауға жасалынған бекітілген құрылғылардың сае алуан бөлшегінің қатысуымен расталады. Маңыздысы:

– Post пен Get ұғымын қалыпты түрде қабылдау, сонымен қатар ертеден расталған массивтерді web-беттің мекеніне айнымалыны қалыптастыру;
– сан алуан мәліметтер қорын басқару системалары;
– HTTP бөлімшелерін қалыпты ретпен тарату;
– Cookies қызмет атқару;
– алыстаға мәліметтерменен, сокеттермен қызмет атқару;
– жүйеге жазылған мәліметтерді зерттеу;

Осы кезде PHP көптеген құрушылармен пайдаланады. Көрсеткіштік TIOBE қалыптастырушысының құжатымен, 01.05.16ж PHP бағдарламалау тілі санының ішінде алтыншы сатыда бағаланды, анықтау системасы мәліметтер расталған. PHP-ді пайдаланатын улкен жобалар Vkontakte, уикипедия т.б. .

Жалпы web жобаларды жасау сонымен қатар hosting-ке программалық кепілдің тобы бекітілген. PHP – серверлік тарапында қызмет атқаратын веб жобасының құрамынан жасалған және бұл дайындаушыларға динамикалық жобаларды тез сонымен қатар сапалы формада жасауға қызмет етеді. PHP – C программалау тіліне жақын және бағдарламашылар он джава және C++ тәлдеріне де жобалас екенін айтады.

Сапалы мүмкіншіліктердің арасынан басым анықтамаларды масивпен қызмет атқарудың мықты құрылғыларды, мәліметтер базасымен қызмет атқарудың көп пайдаланушылығын көру мүмкін.

Веб парақшаны құрудың стандартты характеристикасы пайдаланатын динамикалық бөлшекті істеу мен реттеу маңызды құрылғы сияқты пайлануы да болады. Көптеген қалыпты қасиеттерінің көмегімен PHP құрушының санасына қалыптасқан түрлі қиыншылықты ойластырады. Сан алуан процесстерді қалыптастыруға, тапсырмаларды, түрлі техгологияны жасай алады.

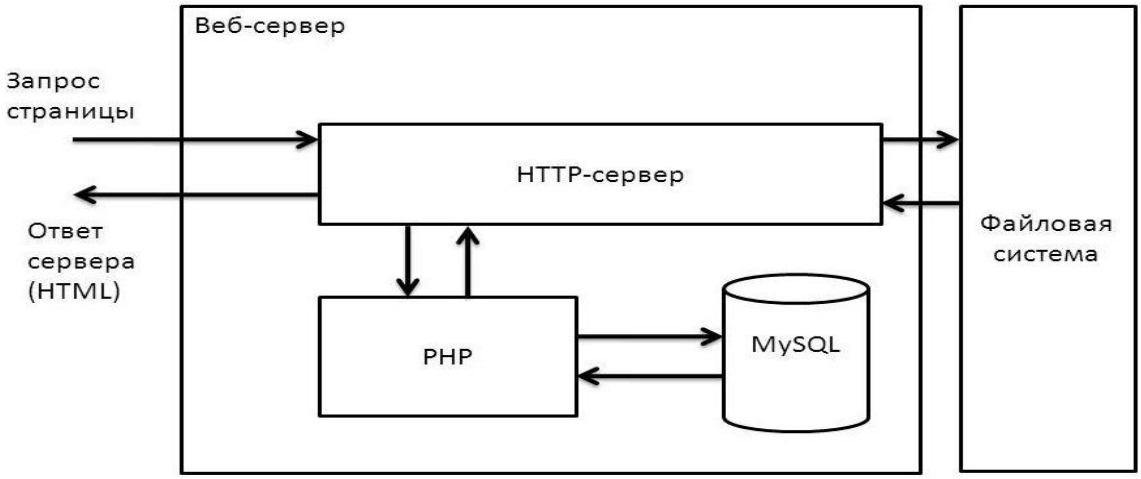
PHP-дегі мумкіншіліктердің көптеген бейтаныс компоненттерді біріктіріп, күрделі қызметтерді құрудан алыстау мүмкін, сол себептен сан алуан бағдарламашылар PHP-ді артық көреді. PHP-дің бұдан бөлек жақсы тұсы ол HTML файлына бекітілетінінде, сол себептен бағдарламашыға қалыпты HTML қоюына сан алуан операторлар қарастырылады, сондықтан бағдарламаны жазбауға болады.

PHP сервер бетінде жүзеге асады да. Содан кейін HTML құжатын дайындайды, яғни ол клиентке барады. Сосын оның қорытындысы ғана бейнеленеді, оның қалай жасалғанын көре алмайды. Қалыпты HTML құжатын PHP қарастыратындай қыла аламыз, ал клиент HTML құжатының қорытындысынан хабарсыз болады.

PHP веб-серверге баратын сктрипт арқасында веб-парақшаның динамикасына керекті программалау тілі.

MySQL жоба дайындағанда пайдалануға ыңғайлы әрі тез мәліметтер

базасының сервері. PHP-де, яғни жобаны жасау барысында скриптпен пайдаланады. PHP мен Деректер қоры көрінісі 2.7-суретте бейнеленген.



2.7-сурет – PHP мен Деректер қоры

phpMyAdmin – PHP-де берілген вем-бағдарлама мен MySQL системасын бақылауға жасалған веб көрініс. phpMyAdmin көмегімен желіні қолданып MySQL-ді реттеуге, осы SQL-дің операторы арқылы жүзеге асыруға, мәліметтер базасындағы жобаларындағы мәліметтерді жасау мүмкін. phpMyAdmin жиі пайдаланылатынының бірден бір түсінігі – SQL командаларын термей осы MySQL-де жеңіл түрде жүзеге асырылады. 2.8-суретте SQL-дің көп қолданылатын командалары көрсетілген.

```
Оларды мынадай топтарға бөлуге болады:
 - Деректерді анықтайтын операторлар
   . CREATE объектіні жасайды
   . ALTER  объектіні өзгертеді
   . DROP  объектіні жояды
 - Деректермен әртүрлі әрекеттер жасайтын операторлар
   . SELECT  берілген шарт бойынша деректерді оқиды
   . INSERT   жаңа деректерді қорға қосады
   . UPDATE  қордағы деректерді өзгертеді
   . DELETE  деректерді жояды
 - Деректерге рұқсат берумен айналысатын операторлар
   . GRANT  пайдаланушыларға объектімен белгілі бір операциялар төңірегінде рұқсатты айқындайды
   . REVOKE  бұрын берілген рұқсаттарды қайтарып алады
   . DENY  берілген рұқсаттың басымдылығын айқындап, оны жоққа шығарады
 - Транзакцияларды басқаратын операторлар
   . COMMIT  транзакцияны қолданады
   . ROLLBACK   транзакцияға ұшыраған өзгерістерді бұрынғы қалпына келтіреді
   . SAVEPOINT  транзакцияны кішкене бөлшектерге бөледі
```

2.8-сурет – SQL командалары

Жалпы айтқанда phpMyAdmin көмегімен мәліметтер базасын, сызбаларын, командаларын бекітуге, олармен жұмыс атқаруға яғни өшіруге, түрлі процесстер барысында көмегін қолдану мумкін.

Жалпы phpMyAdmin жобаны құрушылардың түпкі құрылғысы. Сол себептен осы сайманды терең түсіну қажет және түсіну жеңіл.

SQL – командаларының арасындағы жиі пайдаланатындары:

– SELECT – шығару;
– INSERT – енгізу;
– UPDATE – жаңарту;
– DELETE – жою;

PhpMyAdmin – ол ҚОЮ көмегімен қолданушылардың қызметін оіайлатады, ҚОЮ ссылка көмегімен INSERT командасын атқарады.

Негізгі түпкі мақсаты есептерді шығару үшін жасалынған. Бұл тілді пайдалану үш жалпы салаға тармақталады:

– сервердің қасында жүзеге асатын скрипттерді дайындау;
– желіге тәуелсіз жалпы өз алдына жеке жүзеге асатын скрипттерді дайындау;
– графиктік көріністерді дайындау.

PHP басты 5 сипаттамадан тұрады:

– стандартты;
– жеңіл;
– пайдалы;
– сенімді;
– адаптивті.

Бұл тіл көптеген сферада қызмет атқарған бағдарламашыға белгілі және ұқсас, PHP синтаксисі СИ тіліне келеді.

Қарапайым скриптті жазу үшін, қандай да бір айнамалыларды, модульдерді т.б. сипаттау міндет емес. Кез келген скрипт PHP операторларын басталуына болады. PHP-де Сәлем, Әлем программасының көрінісі 2.9-суретте көрсетілген.

```
<?php
  echo 'Hello, world!';
?>
```

2.9-сурет – PHP коды

Екінші жеңіл түрі де бар, 2.10-суретте берілген.

```
<?= 'Hello, world!' ?>
```

2.10-сурет – 2-ші түрі

PHP “ <? php ?>” осындай шектеулердің құрамында бекітілген кодты жүзеге асырады.

PHP-де нақты кілтті сөздер ұғымы бар, сол кілттік сөздер тізімі 2.11-суретте көрсетілген.

__halt_compiler()	abstract	and	array()	as
break	callable (с PHP 5.4)	case	catch	class
clone	const	continue	declare	default
die()	do	echo	else	elseif
empty()	enddeclare	endfor	endforeach	endif
endswitch	endwhile	eval()	exit()	extends
final	finally (с PHP 5.5)	for	foreach	function
global	goto (с PHP 5.3)	if	implements	include
instanceof	insteadof (с PHP 5.4)	interface	isset()	list()
namespace (с PHP 5.3)	new	or	print	
private	protected	public	require	require_once
static	switch	throw	trait (с PHP 5.4)	
try	unset()	use	var	while

2.11-сурет – Кілттік сөздер тізімі

SublimeText – көптеген редактрлеу функцияларына толы мәтіндік редактор. Мәтіннің толықтай баптауларының көрінісі, түрлі кодтаманың қолдаулары, шексіз артқа жүгіну процетері, GIT-тың қолдаулары, терудің автоматты аяқталуы, бағандарды таңдау мүмкіндеіктері, макрос жүйесі, мәліметтің өз бетінше сақталып отыруы, соңғы әрекеттің қайталау процесі, ыңғайлы үлкен іздеу процесі нәтижесін белгілеп және осындай көптеген мүмкіндіктермен қамтамасыздандырылған. Келесі программалау тілдерінің синтаксисімен қамтамасыздандырылған:

– D;
– JavaScript;
– C;
– PHP;
– C++;
– HTML;
– SQL;
– C#;
– Python;
– CSS;
– Java;
– Perl және т.б. .

SublimeText – дизайны жағымды әрі өте ыңғайлы. Әдеттегі безендіру тақырыбы қараңғы түсте болады, яғни ол көп қолданушыларға үйреншікті бірақ

басқа түске ауыстыруға болады (2.12-суретте SublimeText редакторының логотипі көрсетілген).

2.12-сурет – SublimeText редакторының логотипі

Желіде реттелген html файлдарға негізгі сол мәлімет файлын өзгерістерге ұшыратпай стильді қолдануға, көрсетілуіне мақсатталған және html бөліктерімен толықтыру мүмкінтері бар. Яғни бұл жаңарту түрінің ыңғайлы варианты – CSS жақтайтын желіні кездестіру.

CSS өркендеуді қолданушы өздігінен, қарапайым жолмен редакрлеуде жүзеге асыра алады, алайда бұл құжаттармен қызмет атқаруға Джава немесе осы сияқты программалауда істеу мүмкін. Осы процесті жүзеге асыруға библиотекаларды қолдануға рұқсат.

CSS нысандық моделі программалау библиотекасының ерекшелегі ретінде саналады. Осыны объектімен теңеуге болады, яғни түгел барлық моделі, баптаулары, қасиеттерінің түсінігі бар, алайда негізгі коды жоқ.

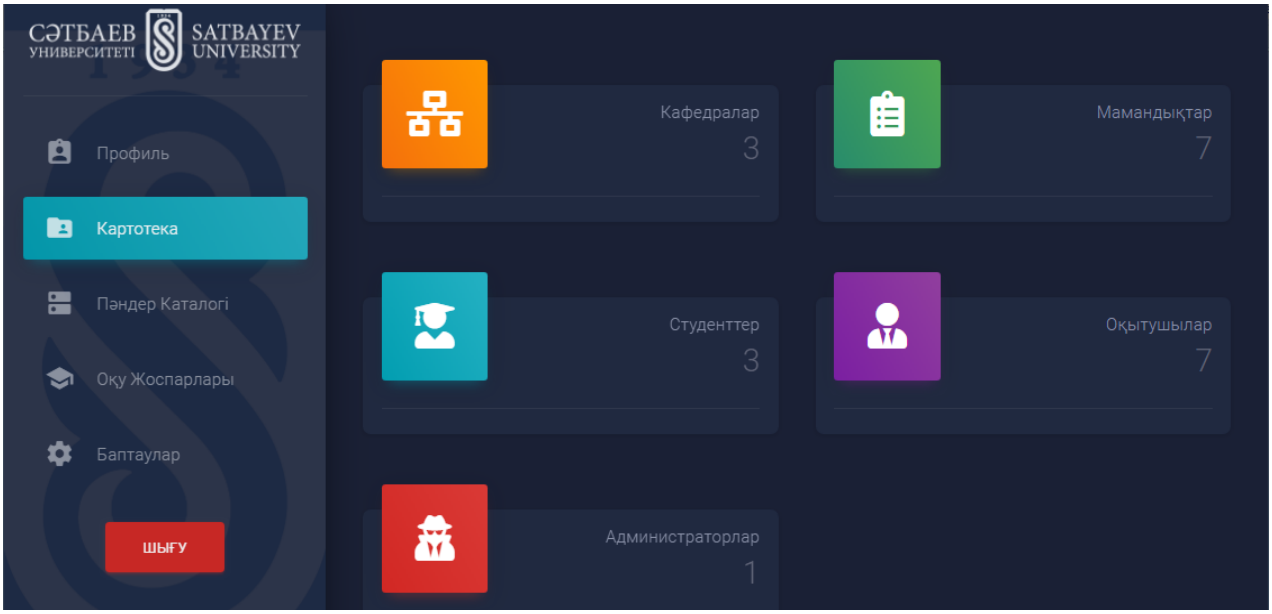
CSS файлы формальды тіл CSS-те беріледі, немесе веб-беттің ішінде сондай-ақ сипаттап отырған беттің сыртында, CSS форматымен берілген бөлек файлда жазылу мүмкін. CSS форматы – ол қарапайым мәтіндік файл.

3 Қолданбалы бөлім

3.1 Жеке оқу жоспарын тіркеуші веб қосымшаны іске қосу

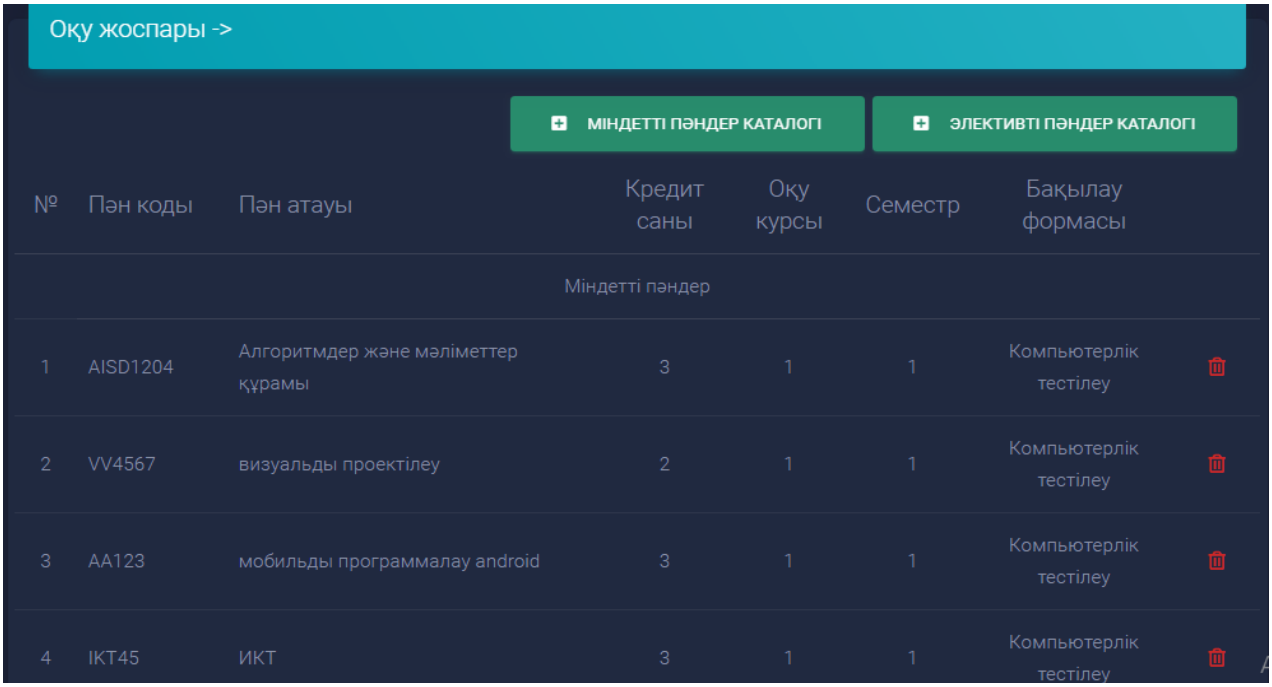
Оқу пәнінің электронды каталогы атауындағы веб-қосымшасы қолданысқа толығымен дайын. Пайдаланушылар осы веб-қосымшаны түрлі орында отырып пайдалануларына болады, ал бұл жобаның қолданатын жандар – білім алушылар. Жоба білім алушыларға міндеттелген процестерді жеңілдетуге мақсатталған. Администратордың беті мен білім алушы беті бар. Алғашқы қадам, админ өз логин-парольін теріп өзінің жеке парақшасына кіреді. Бұл бет қандай парақшалардан тұрады 3.1 – суретте көрсетілген. Admin жеке оқу жоспарын құрмас алдын картотека бетіндегі мына парақшаларды толықтырады:

– кафедралар;
– мамандықтар;
– студенттер;
– оқытушылар.



3.1-сурет – Картотека бөлімі

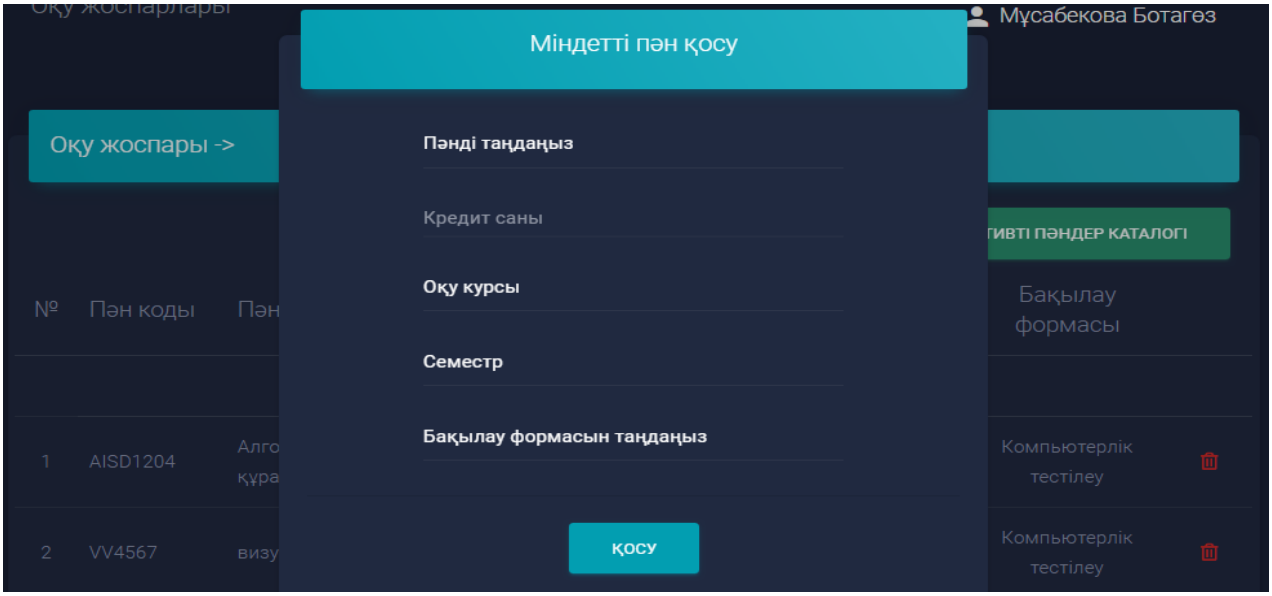
Яғни, бұл парақшаларда қандай кафедралар, мамандықтар, студенттер және оқытушылар тізімі бар толықтай енгізіліп шыгады, сонымен қатар оқытушы қандай сабақтан береді таңдап, бекітіледі. Ал пәндер тізімін таңдар алдын Пәндер каталогы бетіне кіріп белгілі піннің атауын, шифрін, қай кафедраға бағынады көрсетіп, сақтайды. Осы пунктілерді жүзеге асырған соң admin мамандықты енгізіп, сол мамандыққа оқу жоспарын құрады. Оқу жоспарының интерфейсі 3.2-суретте көрсетілген. Оқу жоспарында пәндерді элективті және міндетті пәндерге бөліп, кредиттік санын, оқу курсын, қай семестр, бақылау формалы қалай жүзеге асады енгізеді.



3.2-сурет – Оқу жоспарының құрылымы

3.2 Күзгі және көктемгі оқу семестріне тіркеуді ұйымдастыру

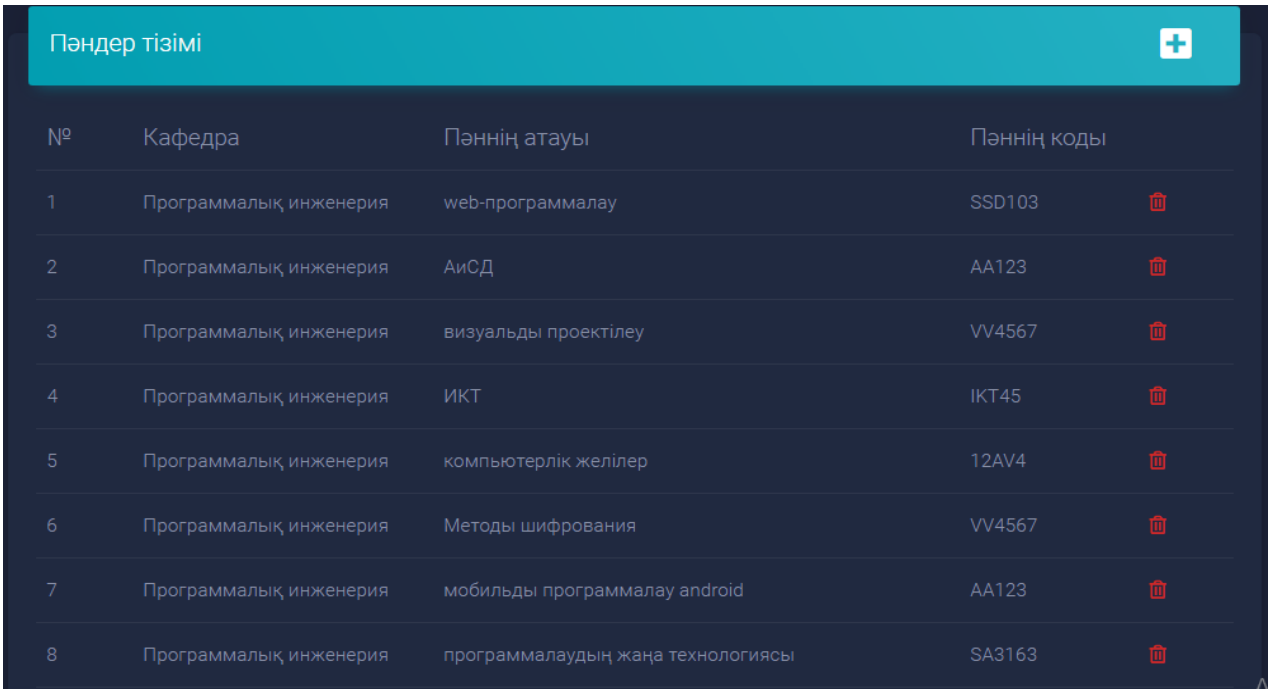
Міндетті пәнді қосу беті 3.3-суретте көрсетілген, және элективті пәндерді қосу реті де міндетті пәнді қосу сияқты қадамдардан тұрады.



3.3-сурет – Міндетті пән қосу

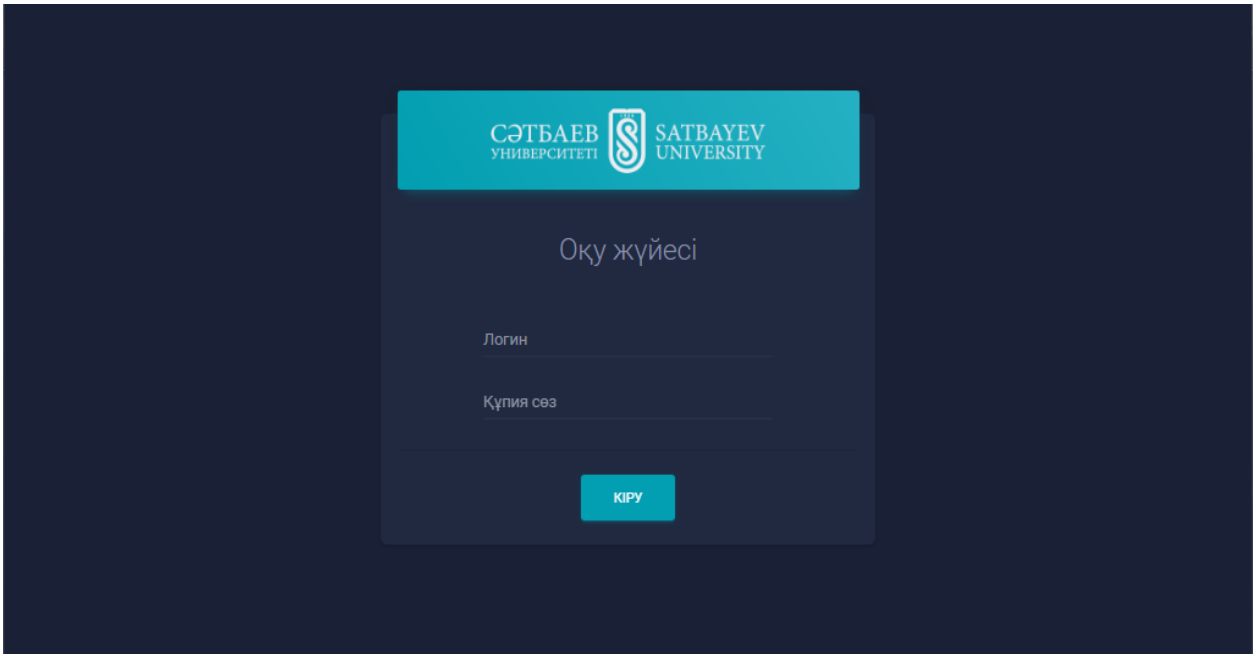
Оқу жоспарындағы басты рөл атқаратын – пәндер тізімі. Әр пәннің атауы

және коды болады, 3.4 суретте қосылған пәндер тізімі бейнеленген. Жаңа пәнді енгізу барысында пәннің қай кафедраға тиісті екені таңдалады, кафедра тізімі беріледі.



3.4-сурет – Пәндер каталогы

Білім алушынынында алғашқы қадамы – жүйеге кіру. 3.5-суретте жүйеге кіру беті көрсетілген.

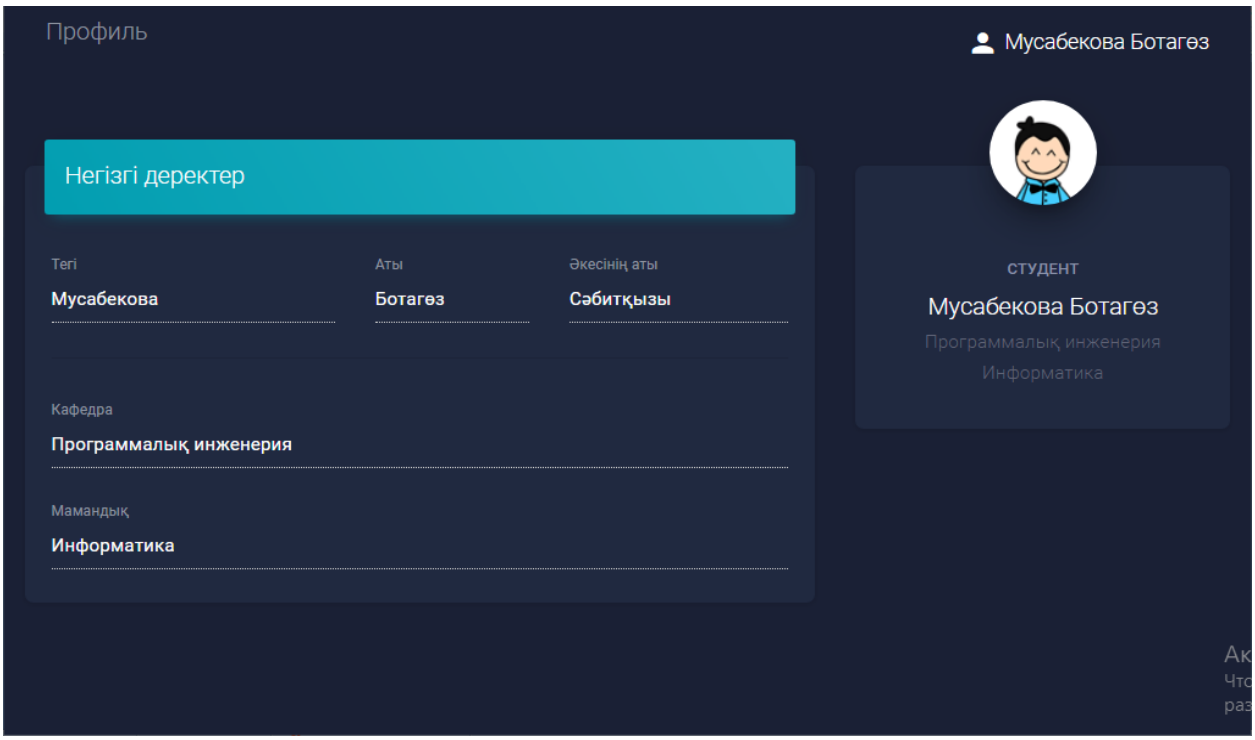


3.5-сурет – Оқу жүйесіне кіру

Білім алушы беті мына парақшалардан тұрады:

– негізгі деректер;
– жеке Оқу Жоспары;
– баптаулар.

Негізгі деректер парақшасында білім алушы туралы қысқаша мәлімет жазылады, яғни аты-жөні, қай кафедра және қай мамандықта окитыны. 3.6-суретте берілген.



3.6-сурет – Негізгі деректер

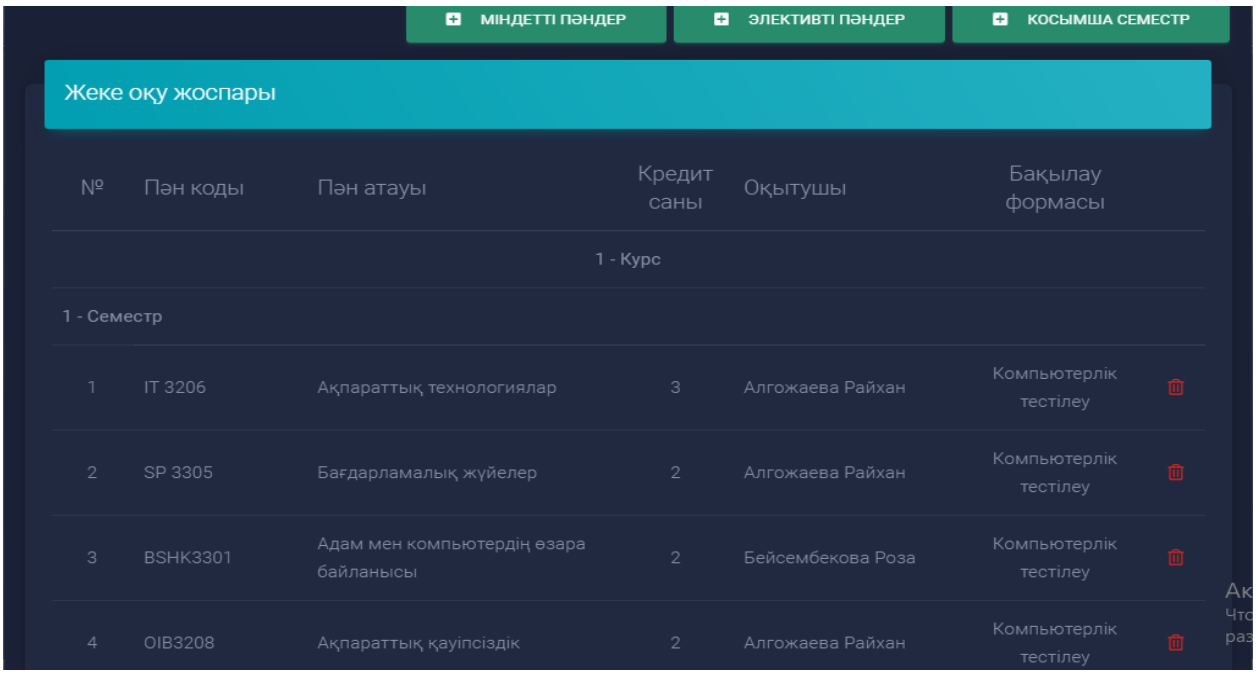
Келесі қадамда Жеке Оқу Жоспары бөліміне өтіп, ЖОЖ құру. ЖОЖ-ның құрылымы суретте берілген. ЖОЖ құру бетінде таңдалатын пәндер түрі:

– міндетті пәндер;
– элективті пәндер;
– қосымша семестр.

Міндетті пәндер бөлімінде студенттің меңгеруіне тиісті пәндер тізімі шығады. Тізім қатарынан пәндерді таңдап, сол пәнді оқытатын оқытушыны бекітеді. Пәнді таңдау уақытында белгілі пән қай семестрде, сонымен қатар қай курста оқылу қажет және неше кредиттік саны бар беріледі.

Элективті пәндерді таңдау бөлімінде де, белгілі пәндер тізімі болады. Құрылымы міндетті пәндерді таңдау бетіне сәйкес.

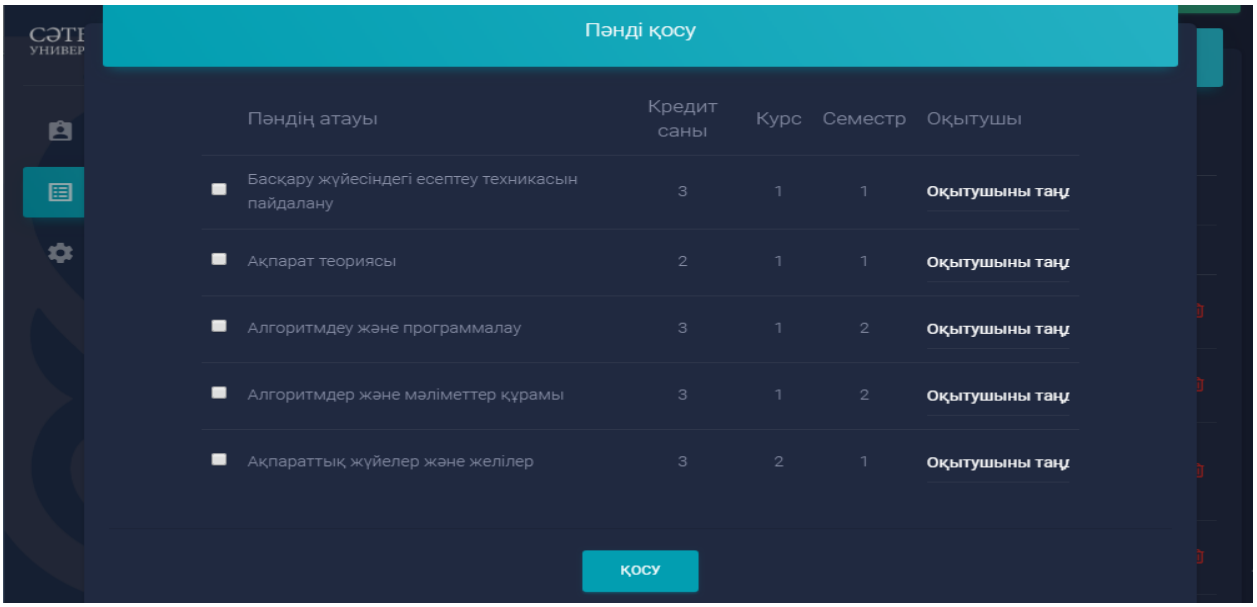
Қосымша семестр бетінде білім алушы қарыз болған пәнді таңдап, игеруді жүзеге асырады. Ал бұл тізімде білім алушы меңгерген пәндер тізімі беріледі, сол пәндер тізімінен қарыз болған пәнді таңдап сақтайды және ол пән ЖОЖ бетінде бекітіледі. ЖОЖ беті 3.7 суретте көрсетілген.



3.7-сурет – ЖОЖ беті

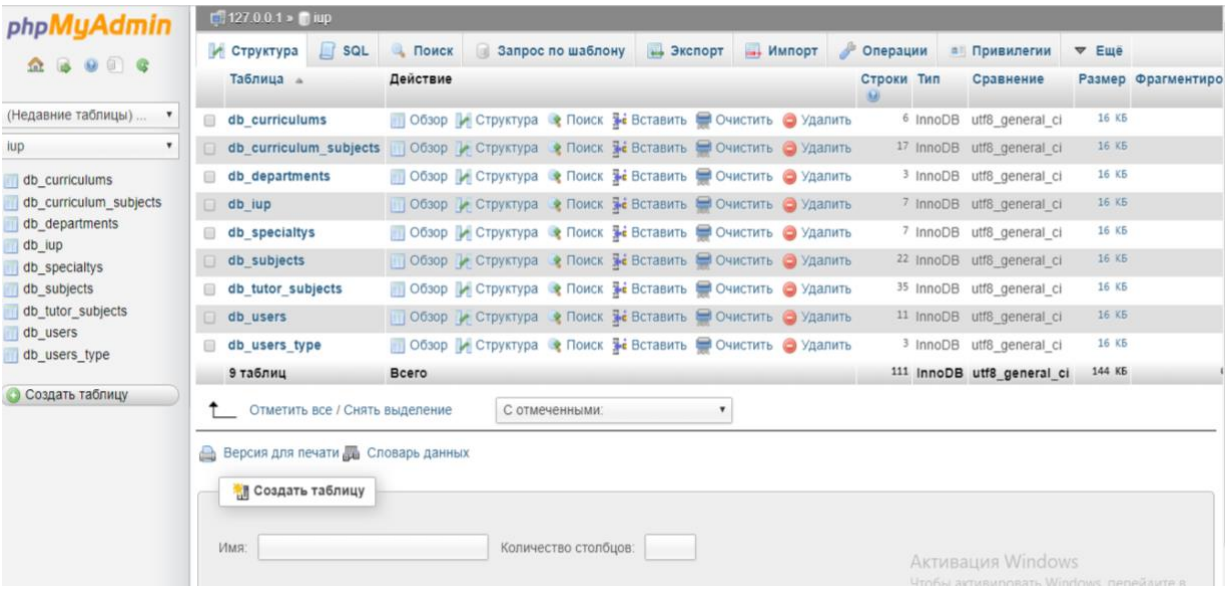
3.3 Академиялық қарыз курстарына тіркелуді ұйымдастыру

Админ бетте құрылған дисциплиналар тізімі бойынша өзінің Жеке Оқу Жоспарын құрады, сонымен қатар Қосымша семестр деген ұғым болады. Бұл парақшада білім алушы меңгерген пәндер тізімі көрсетіледі, білім алушы сол қарыз болған пәнді, оқытушыны таңдап, өзінің Жеке Оқу Жоспарындағы пәндер тізімінің қатарына сақтайды. 3.8-суретте пәнді қосу беті берілген.



3.8-сурет – Пәнді қосу беті

Дипломдық жобаның деректор қорындағы құрылымы. 3.9 суретте көрсетілген. Деректер қорындағы таблицалар: db_curriculums, db_curriculum_subjects, db_departments, db_iup, db_specialtys, db_subjects, db_tutor_subjects, db_users, db_users_type.



3.9-сурет – Деректер Қорындағы таблицалар тізімі

ҚОРЫТЫНДЫ

Қазіргі таңда жоғарғы оқу орындарына арнайы жасалған университет порталдарының саны артып келе жатыр. Бірақ басым көпшілігін қолдану, сонымен қатар түсіну қиындықтар тудыратын да жағдайдар кездесіп жатады. Қосымша семестрді онлайн түрде бекіту міндеті қарастырылмаған. Ал бұл мүмкіндіктерді пайдалану, ауқымды, сондай-ақ студент саны көп жоғарғы оқу орындарында керекті қадамдар болып есептеледі. Жеке оқу жоспарын толығымен қазақша түрде жазып, сан алуан мүмкіншіліктерді пайдаланып, жаңа функцияны енгізу болды.

Бұл дипломдық жобада мақсат етілген жоспарлар мен міндеттер қарастырылып, жүзеге асты:

– веб-қосымшаның құрылымы жасалынды;
– қолайлы әрі ыңғайлы веб-қосымшасының интерфейсі құрылды;
– уақытты тиімді пайдалануды жүзеге асырды;
– қосымша семестрді тез және жеңіл тіркеуге болады;
– жеке оқу жоспарын құру жасалынады.

Жобадағы жүзеге асқан теориялық бөлімдерге қысқаша сипаттама:

Алғашқы бөлімінде білім алушының жеке оқу жоспары, элективті дисциплина, қосымша семестр туралы және осы ұғымдардың мүмкіншіліктері, қолданылуы жайында жазылды, сәйкес веб-қосымшамен өзгешелігі ашық сипатталды.

Келесі бөлімде веб-қосымшаны диаграммалар арқылы сипаттап, жобаны түсіндіру және қолданылған бағдарламау тілі, мәтіндік редакторы, қолданылған визуальды модельдеудің құралына туралы жазылды.

Соңғы бөлімде веб-қосымшаның автоматтандырылуы және жеке оқу жоспарының интерфейсі, мүмкіншіліктерінің скриншоттары көрсетілген т.б.

Қорытындалай келгенде, бүгінгі таңда ақпарттық технологиялар жүйесі тез даму үстінде. Жаңа технологиялар қажетті және адам өміріне ауқымды өзгерістер әкелуде. Ал осы жобада міндеттелген мақсаттар жүзеге асып, оқу пәнінің электронды каталогы толығымен жасалынды.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1 Руководство по составлению индивидуального учебного плана обучающегося (ИУПО) университета СУ // Сайттың электронды нұсқасы http://portal.kaznitu.kz/files/manual_IUP_2015ru

2 Руководство пользователя по регистрации студента на курсы на образовательном портале университета // Сайттың электронды нұсқасы http://portal.kaznitu.kz/files/IUPru

3 Каталог элективных дисциплин // Сайттың электронды нұсқасы https://www.do.ektu.kz/doektu/files/catalog

4 Проектирование программного обеспечения Разработка веб-сайтов, UML Design // Сайттың электронды нұсқасы https://habr.com/ru/post/74330/

5 Ташков П. – Веб мастеринг на 100%: HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрутка. – СПб.: Питер, 2010. – 512 с.

6 Робин Никсон – Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5**. –** Питер**:** 2016. – 768 с.

7 Иванов И.Н. Веб промышленного пред.приятия. Учебник. – М.: ИНФРА–М, 2011. – 395 с.

8 Мамаев Е. MySQL Server. Проектирование и реализация баз данных. Сертификационный экзамен. – СПб.: BHV, 2004. – .416 с.

9 Кватрани, Т. Rational Rose 2000 и UML. Визуальное моделирование: Книга по Требованию, 2009. – 176 с.

10 Боггс, М. – UML и Rational Rose – Москва: 2010. – 260 с.

11 Веллинг, Л. – Разработка веб-приложений с помощью PHP и MySQL, 2016. – 794 с.

12 Томсон Л. – Разработка Web-приложений на PHP и MySQL: пер. с англ. – Лаура Томсон, Люк Вел-линг, 2003. – 672 с.

13 Б. Хеник – HTML и CSS. Путь к совершенству HTML и CSS: The Good Parts, 2010. – 500 с.

14 К. Шмитт – CSS. Рецепты программирования, 2014. – 462 с.

15 Комолова, Н. – HTML. Книга по Требованию, 2011. –288 с.

16 Мак-Дональд М. – HTML5. Недостающее руководство – Петербург, 2012. – 480 с.

А Қосымшасы
(міндетті)

«Оқу пәнінің электронды каталогы» web-қосымшасын құруға арналған техникалық тапсырма

А.1 Кіріспе

Жаңа технологиялар адамның күнделікті өміріне қажетті және уақытты тиімді қолдануға мүмкіндік береді. Кредиттік технологиямен оқыту барысында білім алушының жеке оқу жоспары құрастырылады. Жоғарғы оқу орнының қабырғасында білім алушының өзінің қарамағына алатын алғашқы құжаттардың бірі – Жеке Оқу Жоспары.

Жеке оқу жоспары – элективті дисциплиналар каталогы және оқу жоспары негізінде группа жетекшісінің көмегімен, студенттің әрбір жылдық оқу жылына арнайы, өзі толықтай құрастыратын пәндер тізгіні. Білім алушы ЖОЖ бекітуге және оқу жоспарының шарттарына сай берілген пәндерді меңгеруге жауапты. Жоғарғы оқу орнының порталында жеке оқу жоспарын құрудың өз ережелері мен қадамдары болады.

А.1.1 Жобаның мақсаты мен міндеті

Жоба білім алушының жұмыс оқу жоспарын, жеке оқу жоспарын электронды түрде оқу порталында жүзеге асырады. Сонымен қатар, жеке оқу жоспары бөлімінде қосымша семестрге тіркелу операциясы электронды түрде орындалады. Электронды түрде қосымша (жазғы) дисциплинаға тіркелу, білім алушылар мен оқытушыларға, сонымен қатар тіркеуші бөлім қызметкерлеріне өте тиімді технология.

А.1.2 Қолдану саласы

Оқу пәнінің электронды каталогын жеке оқу жоспарын, сонымен қатар қосымша семестрді бекітетін студенттер ортасы пайдаланады.

А.1.3 Анықтамалар, терминдер және қысқартулар

PHP веб-серверге баратын сктрипт арқасында веб-парақшаның динамикасына керекті программалау тілі.

MySQL жоба дайындағанда пайдалануға ыңғайлы әрі тез мәліметтер базасының сервері.

UML (ағыл. Unified Modeling Language – бірыңғай модельдеу тілі) – бұл графикалық сипаттама тілі Бағдарламаның нысаны сипатын әзірлеу үшін пайдаланылады. UML-моделы болып аталынатын графикалық символдарды пайдаланып жүйенің абстрактілік модельін құру. UML бағдарламалау тілі болып табылып, UML модельдері негізінде кодты генерациялауы мүмкін.

SublimeText – көптеген редактрлеу функцияларына толы мәтіндік редактор.

Денвер – қазіргі уақыттағы танымал локальдық серверлердің бірі және бұл локальдық сервердің беретін мүмкіншілігі сайт жасағанда желіге қосылуды қажет етпей етпейді.

А.2 Жүйенің жалпы сипаттамасы

А.2.1 Оқу пәнінің электронды каталогы жалпы сипаттамасы

Дипломдық жобаны құру үшін келесі тапсырмалар орындалады:

– веб-қосымшаның құрылымы жасалынды;
– қолайлы әрі ыңгайлы веб-қосымшасының интерфейсі құрылды;
– уақытты тиімді пайдалануды жүзеге асырды;
– қосымша семестрді тез және жеңіл тіркеуге болады;
– жеке оқу жоспарын құру жасалынады.

А.2.2 Пайдаланушылық интерфейстер

Қолданушы жүйеге кіргенде онда қысқаша мәлімет пен суретін көруге ыңғайлы және қарапайым етіп жасалған интерфейс құрастырылған. Картотека бөлімінде бес бөлімшелер бар. Әр бөлімшеде жеке оқу жоспарын құратын керекті элементтер толтырылады. Бөлімшелерді қосқан соң, соңғы қадамда жеке оқу жоспарын бекітетін бөлім орналасқан.

Б Қосымшасы
(міндетті)

Бағдарлама мәтіні

1. Жүйеге логин мен құпия сөз арқылы кіру бетінің мәтіні

```
<?php
session_start();
include('config.php');
if(isset($_POST['login_test'])){
  $login     = $_POST['login'];
  $password  = $_POST['password'];
  $login    = stripslashes($login);
  $login    = htmlspecialchars($login);
  $login    = trim($login);
  $password = stripslashes($password);
  $password = htmlspecialchars($password);
  $password = trim($password);
  $password = md5($password);
  $query  = mysql_query("SELECT * FROM db_users WHERE login='$login'
AND password='$password' ",$db);
  $result = mysql_fetch_array($query);
  if($result['user_id'] == '') header('Location:./?error');
  else if($result['user_id'] != ''){
    $_SESSION['login_iup'] = $result['login'];
    $_SESSION['iup_user_type'] = $result['user_type'];
    header('Location:./');
  }
}
?>
```

2. Профиль бетінің мәтіні

```
<div class="row">
  <div class="col-md-8">
    <div class="card">
      <div class="card-header card-header-info">
        <h4 class="card-title">Негізгі деректер</h4>
      </div>
      <div class="card-body">
```

```
        <br>
        <div class="row">
          <div class="col-md-5">
            <div class="form-group">
              <label class="bmd-label-floating">Тегі</label>
              <input type="text" class="form-control" disabled
value="<?=$result_user_info[user_sname]?>">
            </div>
          </div>
          <div class="col-md-3">
            <div class="form-group">
              <label class="bmd-label-floating">Аты</label>
              <input type="text" class="form-control" disabled
value="<?=$result_user_info[user_name]?>">
            </div>
          </div>
          <div class="col-md-4">
            <div class="form-group">
              <label class="bmd-label-floating">Әкесінің аты</label>
              <input type="email" class="form-control" disabled
value="<?=$result_user_info[user_mname]?>">
            </div>
          </div>
        </div>

        <hr>
        <br>

        <?php
        if($_SESSION['iup_user_type'] != 1){

          echo "
                <div class='row'>
                  <div class='col-md-12'>
                    <div class='form-group'>
                      <label class='bmd-label-floating'>Кафедра</label>
                      <input type='text' class='form-control' disabled
value='$result_user_info[department_name]'>
                    </div>
                  </div>
                </div>
```

```
                <br>

                <div class='row'>
                  <div class='col-md-12'>
                    <div class='form-group'>
                      <label class='bmd-label-floating'>Мамандық</label>
                      <input type='text' class='form-control' disabled
value='$result_user_info[name_specialtys]'>
                    </div>
                  </div>
                </div>";
        }
        ?>

      </div>
    </div>
  </div>
<div class="col-md-4">
    <div class="card card-profile">
      <div class="card-avatar">
        <img class="img"
src="assets/img/faces/<?=$result_user_info[user_ava_url]?>" />
      </div>
      <div class="card-body">
        <h6 class="card-category"><?=$result_user_info[type_name]?></h6>
        <h4 class="card-title"><?=$result_user_info[user_sname]."
".$result_user_info[user_name]?></h4>
        <p class="card-description">
          <?php
          if($_SESSION['iup_user_type'] != 1) echo
$result_user_info[department_name]."<br>".$result_user_info[name_specialty
s];
          ?>
        </p>
      </div>
    </div>
  </div>
</div>
```

3. Пәндер каталогы бетіне жаңа пәнді енгізу мәтіні

```
<?php
  if(isset($_POST['insert_new_subject'])){

    $name_subject = $_POST['name_subject'];
    $kod_subject = $_POST['kod_subject'];
    $department_id = $_POST['department_id'];

    $insert_new_subject = mysql_query(" INSERT INTO db_subjects
(name_subject, kod_subject, department_id)
                                        VALUES ('$name_subject', '$kod_subject',
'$department_id') ");

    if( $insert_new_department == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?subjects\">");

  }else if(isset($_GET['del'])){

    $delete_subject = mysql_query(" DELETE FROM db_subjects WHERE
id_subject = '$_GET[del]' ");

    if( $delete_subject == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?subjects\">");

  }

  ?>
```

4. Жаңа мамандық енгізу мәтіні

```
<?php
if(isset($_POST['insert_new_specialty'])){

  $specialtys_name = $_POST['specialtys_name'];

  $insert_new_specialty = mysql_query(" INSERT INTO db_specialtys
(name_specialtys, department_id) VALUES ('$specialtys_name',
'$_POST[department_id]') ");
```

```
if( $insert_new_department == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?cardfile=departments\">");

}else if(isset($_GET['del'])){

  $delete_specialty = mysql_query(" DELETE FROM db_specialtys WHERE
id_specialty = '$_GET[del]' ");

  if( $delete_specialty == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?cardfile=spesialtys\">");

}

?>
```

5. Оқытушы енгізу мәтіні

```
<?php
if(isset($_POST['insert_tutor_subject'])){

  $insert_subject_list_tutor = mysql_query(" INSERT INTO db_tutor_subjects
(user_id, id_subject) VALUES ('$_GET[tutor_id]', '$_POST[id_subject]') ");

  if($insert_subject_list_tutor == true) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?cardfile=tutors\">");

}else if(isset($_POST['insert_new_tutor'])){

  $user_sname = $_POST['user_sname'];
  $user_name = $_POST['user_name'];
  $user_mname = $_POST['user_mname'];

  $department_id = $_POST['department_id'];

  $login = $_POST['login'];
  $password = md5("12345");
  $insert_new_student = mysql_query(" INSERT INTO
  db_users (login, password, user_sname, user_name, user_mname,
```

```
user_ava_url, department_id, user_type)
VALUES ('$login', '$password', '$user_sname', '$user_name', '$user_mname',
'no_ava.png', '$department_id', '3') ");
  if( $insert_new_student == true) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?cardfile=tutors\">");

}else if(isset($_GET['del'])){

  $delete_student = mysql_query(" DELETE FROM db_users WHERE user_id
= '$_GET[del]' ");

  if( $delete_student == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?cardfile=tutors\">");

}

?>
```

6. Жеке оқу жоспары бетінің мәтіні

```
<?php
  if(isset($_POST['insert_new_iup'])){

    $i = 0;

    while($i < $_POST['col']){
      $i++;

      $subject = 'subject'.$i;
      $tutor_id = 'tutor_id'.$i;
      $subject_radio = 'subject_radio'.$i;
      $tutor_id1 = 'tutor_id1_'.$i;
      $tutor_id2 = 'tutor_id2_'.$i;

      if( $_POST[$subject] != '' && $_POST[$tutor_id] != '' || $_POST[$subject]
!= '' && $_POST[$tutor_id1] != '' || $_POST[$subject] != '' &&
$_POST[$tutor_id2] != '' ){
```

Б Қосымшасының жалғасы

```
  $query_curriculum_info = mysql_query(" SELECT * FROM
db_curriculum_subjects WHERE id_curriculum_subjects =
'$_POST[$subject]' ");
 $result_curriculum_info = mysql_fetch_array($query_curriculum_info);

if(isset($_POST['summer'])){

        $query_course_of_study = mysql_query(" SELECT
MAX(i.course_of_study) as max FROM db_iup i WHERE i.student_login =
'$_SESSION[login_iup]' ");
        $result_course_of_study = mysql_fetch_array($query_course_of_study);

        $course_of_study = $result_course_of_study['max'];

        $insert_iup = mysql_query(" INSERT INTO db_iup
(id_curriculum_subjects, student_login, id_subject, tutor_id, credits,
course_of_study, semester, form_of_control)
                                        VALUES ('$_POST[$subject]',
'$_SESSION[login_iup]', '$result_curriculum_info[id_subject]',
'$_POST[$tutor_id]', '$result_curriculum_info[credits]', '$course_of_study', '3',
'$result_curriculum_info[form_of_control]' ) ");

       }else{

        if(isset($_POST['elective'])){

          $subject_radio = 'subject_radio'.$i;

          if(isset($_POST[$subject_radio]) && $_POST[$subject_radio] != ''){

            $id_subject = $_POST[$subject_radio];

            if($_POST[$tutor_id1] != '') $tutor_id = $_POST[$tutor_id1];
            else if($_POST[$tutor_id2] != '') $tutor_id = $_POST[$tutor_id2];

          }else exit("<meta http-equiv=\"refresh\" content=\"0;url=./?iup\">");

        }else{

          $tutor_id = $_POST[$tutor_id];
          $id_subject = $result_curriculum_info['id_subject'}
```

```
 $insert_iup = mysql_query(" INSERT INTO db_iup (id_curriculum_subjects,
student_login, id_subject, tutor_id, credits, course_of_study, semester,
form_of_control)
                                        VALUES ('$_POST[$subject]',
'$_SESSION[login_iup]', '$id_subject', '$tutor_id',
'$result_curriculum_info[credits]', '$result_curriculum_info[course_of_study]',
'$result_curriculum_info[semester]',
'$result_curriculum_info[form_of_control]' ) ");

        }
      }
    }

    exit("<meta http-equiv=\"refresh\" content=\"0;url=./?iup\">");
  }else if(isset($_GET['del'])){

    $delete_iup = mysql_query(" DELETE FROM db_iup WHERE id_iup =
'$_GET[del]' ");

    if( $delete_iup == true ) exit("<meta http-equiv=\"refresh\"
content=\"0;url=./?iup\">");
  }
?>
```