

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Программалық инженерия»

Исаева Г.М.

Персоналды тестілеуді автоматтандыруға арналған ақпараттық жүйе өңдеу

ТҮСІНДІРМЕ ЖАЗБА

дипломдық жобаға

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

Алматы 2019

ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Программалық инженерия»

ҚОРҒАУҒА ЖІБЕРІЛДІ

ПИ кафедра меңгерушісі

кант.техн. наук, доцент

ассистент-профессор

 Р. Юнусов

" 13 " май 2019ж.

ТҮСІНДІРМЕ ЖАЗБА

дипломдық жобаға

Тақырыбы: “ Персоналды тестілеуді автоматтандыруға арналған ақпараттық жүйе өңдеу”

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

Орындаған:

Исаева Г.М.

Ғылыми жетекші

лектор

 Ж.Алибиева

" 02 " 05 2019 ж.

Алматы 2019

СӘТБАЕВ УНИВЕРСИТЕТІ

Ақпараттық және телекоммуникациялық технологиялар институты

Кафедра «Компьютерлік және программалық инженерия»

5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»

БЕКІТЕМІН

ПИ кафедра меңгерушісі,

кант.техн. наук, доцент

ассистент-профессор

 Р. Юнусов

"13" маусым, 2019ж.

Дипломдық жобаны орындауға

ТАПСЫРМА

Білім алушыға Иаева Гүлдана Мейірманқызына

Тақырыбы: Персоналды тестілеуді автоматтандыруға арналған ақпараттық жүйе өңдеу

Университет ректоры бұйрығының № 1162-б "16" қазан 2018 ж. шешімімен бекітілген.

Орындалған жобаның өткізу мерзімі "14" мамыр 2019 ж.

Дипломдық жобаның бастапқы мәліметтері: Жобаның төлқұжаты, технология бойынша техникалық құжаттама, техникалық тапсырма, ER-диаграмма түрінде ақпаратты сақтауға арналған деректер қорының сипаттамасы.

Есеп – түсініктеме жазбаның талқылауға берілген сұрақтардың тізімі:

a) пайдаланушы рөлдерін бөлу жүйесін жобалау

b) тұжырымдамаға сәйкес веб-қосымшаның көп деңгейлі архитектурасын енгізу;

в) пайдаланушы интерфейсін жобалау және дамыту;

г) бағдарламаны құру, қателіктерді анықтау және түзету, тестілеу;

Графикалық материалдар тізімі (міндетті суреттердің нақты көрсетілуімен):

презентацияның 20 слайды ұсынылған.



Ұсынылған негізгі әдебиеттер: 14 әдебиеттер тізімінен

Дипломдық жобаны орындау

КЕСТЕСІ

Бөлімдердің атаулары, зерттелген мәселелердің тізімі	Ғылыми жетекшіге және кеңесшілерге ұсыну мерзімі	Ескерту
1. Кіріспе	12.12.2018	МОҚ
2. Тапсырма қойылымы және бағдарламалау ортасын таңдау	16.01.2019	МОҚ
3. Бағдарламалық қамтаманы жобалау.	08.02.2019	МОҚ
4. Бағдарламаны әзірлеу	14.02.2019	МОҚ
5. Бағдарламаның мүмкіндіктерін дамыту	15.03.2019	МОҚ
6. Дипломдық жобаға түсіндірме жазба жазу	30.03.2019	МОҚ

Дипломдық жұмыс бөлімдерінің кеңесшілерінің аяқталған жұмысқа қойған қолтаңбалары

Бөлімдер атауы	Кеңес берушілер (аты-жөні, тегі, ғылыми дәрежесі, атағы)	Қолтаңба қойылған мерзімі	Қолы
Нормалық бақылаушы	А.Б.Рамазан тьютор	13.05.19	
Бағдарламалық бөлім	А.М.Байгаринов техн.ғыл.магистр, лектор	13.05.19	

Ғылыми жетекші  Ж.Алибиева.

Тапсырманы орындауға қабылдап алған студент  Исаева Г.М.

Күні « 08 » 01 2018ж.

АҢДАТПА

Қазіргі уақытта қызметкерлерді тестілеу үшін пайдаланылуы мүмкін көптеген бағдарламалық құралдар бар. Фармацевтикалық компанияның ерекшелігін ескере отырып, функционалдылық пен шектеулерге сай келетін және қауіпсіздік мәселелеріне сәйкес келетін осындай құрал табылған жоқ. Сондықтан өзіңіздің бағдарламалық жасақтамаңызды құру туралы шешім қабылданды.

Сондықтан, тезис «Аптека» фармацевтикалық компаниясында жұмысқа қабылдаудың үлгісі бойынша персоналды іріктеу және сынау жүйесін талдауға арналған. Осы дипломдық жұмыста біз мәселенің теориялық аспектісін зерттеп, кәсіпорынның экономикалық белсенділігін және сыртқы және ішкі ортасын талдаудан бастауды жөн көрдік. Сондай-ақ персоналдық аудит жүргізілді, нәтижесінде проблемалық учаскелер анықталды. Белгіленген мәселелер бойынша жұмысқа қабылдау үшін тест сұрақтары дайындалды.

АННОТАЦИЯ

В настоящее время существует большое количество программных средств, которые можно использовать для тестирования персонала. Учитывая специфику фармацевтического предприятия, не было найдено такое средство, которое бы подходило по функциональным возможностям и ограничениям и соответствовало бы соображениям безопасности. Поэтому было принято решение создать свой собственный программный продукт.

Поэтому дипломная работа посвящена проведению анализа системы отбора и тестирования персонала, на примере принятия работы на фармацевтическое предприятие «Аптека». В данной работе был изучен теоретический аспект данного вопроса и проведен анализ хозяйственной деятельности предприятия и анализ внешней и внутренней его среды. Также был проведен кадровый аудит, в результате которого были выявлены проблемные зоны. По выявленным проблемам были подготовлены тестовые вопросы для принятия на работу.

ANNOTATION

Currently there are a large number of software tools that can be used to test personnel. Given the specifics of the pharmaceutical company, no such tool was found that would be suitable for functionality and limitations and consistent with security considerations. Therefore, it was decided to create your own software product.

Therefore, the thesis is devoted to the analysis of the system of selection and testing of personnel, on the example of taking work at the pharmaceutical company "Pharmacy". In this paper, we studied the theoretical aspect of this issue and analyzed the economic activity of the enterprise and the analysis of its external and internal environment. A personnel audit was also conducted, as a result of which problem areas were identified. On the identified problems were prepared test questions for admission to work.

МАЗМҰНЫ

Кіріспе	8
1 Персоналды тестілеудің теориялық аспектілері	9
1.1 Персоналды тестілеудің негізгі аспектілері	9
1.2 Жұмысқа қабылдау критерилері мен моделдері	12
1.3 Фармацевтикалық аймақтағы кадрлардың резервін қалыптастыру ерекшеліктері	13
1.4 Отандық және шетелдік ұйымдардың тәжірибелеріне сүйеніп персоналды іріктеп таңдау	14
2 «Дәріхана» АҚ қызметтік жүйе орталығының талдануы	15
2.1 «Дәріхана» АҚ жалпы сипаттамасы	15
2.2 «Дәріхана» АҚ персоналды тестілеу жүйесінің тағайындалуы	15
2.3 Персоналды тестілеу жүйесінің қысқаша сипаттамасы	15
2.4 Программалық қамтаманы қолдану және функционалдау ортасының ерекшелігі	16
2.5 Есептің қойылымы және қолдану аймағы	17
3 Персоналды тестілеу жүйесін жобалау және программалық қамтамасын өңдеу	18
3.1 Программалық қамтаманы жобалау	18
3.2 Rational Rose аспабымен жұмыс	20
3.3 Прецеденттер диаграммасы	22
3.4 Тізбектер диаграммасы	23
3.5 Кооперация диаграммасы	26
3.6 Кластар диаграммасы	27
3.7 Күй диаграммасы	28
3.8 Программалық қамтама өңдеу	29
Қорытынды	36
Пайдаланылған әдебиеттер тізімі	37
А қосымшасы. Техникалық тапсырма	38
Б қосымшасы. Программа листингі	40
Ерекшеліктері	49

\

КІРІСПЕ

Қазіргі нарықтық бәсекелестік шеңберінде қызметкерлерді сауатты түрде таңдау немесе іріктеу және кейінгі оларды сол жұмысқа бейімдеу шағын кәсіпорындардағы бәсекелестік пен экономикалық жағдайды анықтайтын жетекші факторлардың бірі болып табылатынын құпия емес.

Қазіргі кезде талаптарды қанағаттандыратын, ұйымға барынша көп пайда әкелетін, белсенді кадрларды іріктеп таңдау әдістеріне өту барлық ұйымдарда қолданылатын шаралар ретінде қарастырылуда.

Фармацевтика өнеркәсібінде соңғы уақытта бос орындардың белсенді өсуі байқалды, соның арқасында көптеген басқарушылар мамандарды іріктеп таңдау қиындықтарына тап болды, бұл өз кезегінде іріктеп таңдау жүйесінің тиімділігін жоғалтуға әкелді.

Кәсіби персоналдардың тапшылығы мәселесі барлық дәріхана желілері үшін өте қажет. Егер компания жұмысқа өтінім берген кез келген адамды қабылдап, тек ресми белгілерге (жасы, білім, жұмыс тәжірибесі) сүйенсе, онда қызметкерлердің сапасы туралы айтудың қажеті жоқ болар еді: оларды жұмыстан босату мәселері орын алып, зиян келтірелі мәлім.

Дипломдық жобаның мақсаты жұмысқа кадрларды іріктеп таңдау жүйесінің тиімділігін арттыру бойынша ұсыныстар мен тест тапсырмаларын әзірлеу болып табылады.

Осы мақсатқа жету үшін төмендегідей тапсырмаларды орындауымыз керек:

- персоналды теориялық таңдауды қарастыру;
- осы мәселе бойынша отандық және шетелдік компаниялардың тәжірибесін талдау;
- өндірістік кәсіпорынның сипаттамасын беру;
- кәсіпорында кадрларды іріктеп таңдау жүйесін талдау;
- кәсіпорында кадрларды іріктеп таңдау жүйесін жетілдіру жолдарын ұсыну.

Осы дипломдық жобаның зерттеу нысаны «Аптека» АҚ қызметкерлерін іріктеу, тестілеу және бағалау жүйесі болып табылады.

Дипломдық жобаның пәні: «Аптека» АҚ-да кадрларды іріктеу, тестілеу және бағалаудың негізгі элементтерінің ерекшеліктері болып табылады.

1 Персоналды тестілеудің теориялық аспектілері

1.1 Персоналды тестілеудің негізгі аспектілері

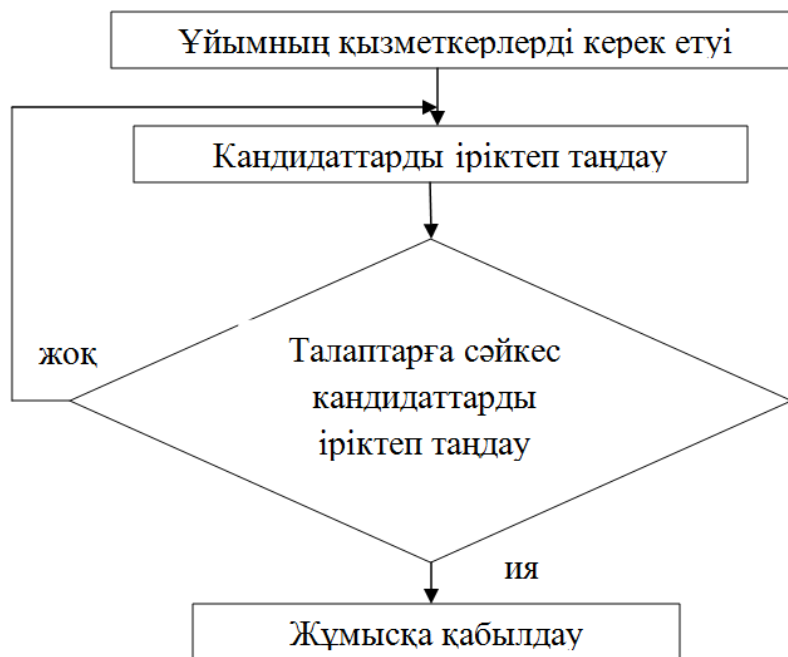
Кесте түрінде берілген персоналды тестілеу түсініктемесінің бірнеше авторлардың ұсыныстары бойынша қарастырылған анықтамаларын салыстыру жұмыстарын жүргізіп өтейік:

Кесте-1 – «Персоналды тестілеу» анықтамалары

№	Анықтамасы	Авторы
1	Бір немесе бірнеше үміткерлерді бос ваканттық орындарға орналастыруға байланысты, оған өтініш білдіргендердің жалпы санынан алумен байланысты процесс. Бұл критерийлерге сәйкес келетін қызметкерлердің жалпы санын бағалаудың белгілі бір әдістерін пайдалана отырып, белгіленген критерий бойынша біреуді таңдау процесі. [5, 10-бет].	А.Я. Кибанов, И.Б. Дуракова
2	Персоналды таңдау - бұл бос ваканттық орындарға үміткер адамдардың жалпы саны арасында бір немесе бірнеше тестілеуден өткен сәйкес келетін кандидаттарды бөлумен байланысты таңдау процесінің кезеңі. [3, 8-бет]	В.В. Владимирова
3	Ұйымның адамның бастапқыда берілген талаптарға сәйкестігін анықтау; сандық және сапалық сипаттамалары, ұйымның мақсаттары мен міндеттеріне сәйкес келетін персоналдың сәйкес құрамын қалыптастыруды қамтамасыз ететін шаралар жүйесі; Ұйымдағы бос ваканттық орындарға сәйкес келетін ең жақсы кандидаттардың бірін таңдау процесі. [2]	Т.Ю. Базарова

Осылайша, берілген анықтамаларды талдау арқылы біз, келесі қорытындыға келдік, берілген ваканттық орындарға персоналды таңдау процессін жақсарту қазіргі кезде маңызды әрі актуальды тақырыптардың бірі болып табылады.

С.В.Шекшн персоналды іріктеп таңдау жүйесінің келесі моделін қолдануды ұсынады (сурет 1).



1-сурет – С.В.Шекшн ұсынған персоналды іріктеп таңдау жүйесінің моделі

Іріктеп таңдау жүйесінің моделі бірнеше кезеңнен тұрады:

а) талаптарды қалыптастыру кезеңі. Жарнамалар берген кезде қабылдау ашрттарын дұрыс беру маңызды болып табылады. Себебі осы кезеңде көптеген іріктеу жұмыстары жүргізіледі. Жұмыс іздеушілердің өзідері ақ талаптарға сай ол жұмысты істей алама жоқ па екендігін түсінеді. Сол себепті осы кезең жұмысы іріктеушілерді дұрыс бағыттау үшін, дұрыс талаптармен берілуі керек.

б) телефонмен сөйлесіп сұхбаттар арқылы қабылдау жұмысы кезінде, арнайы менеджер немесе хатшы белгілі бір критерилер арқылы жұмысқа қалданушыларды іріктеп таңдау жұмысын жүргізі алу қабілеттілігінің кезеңі болып табылады.

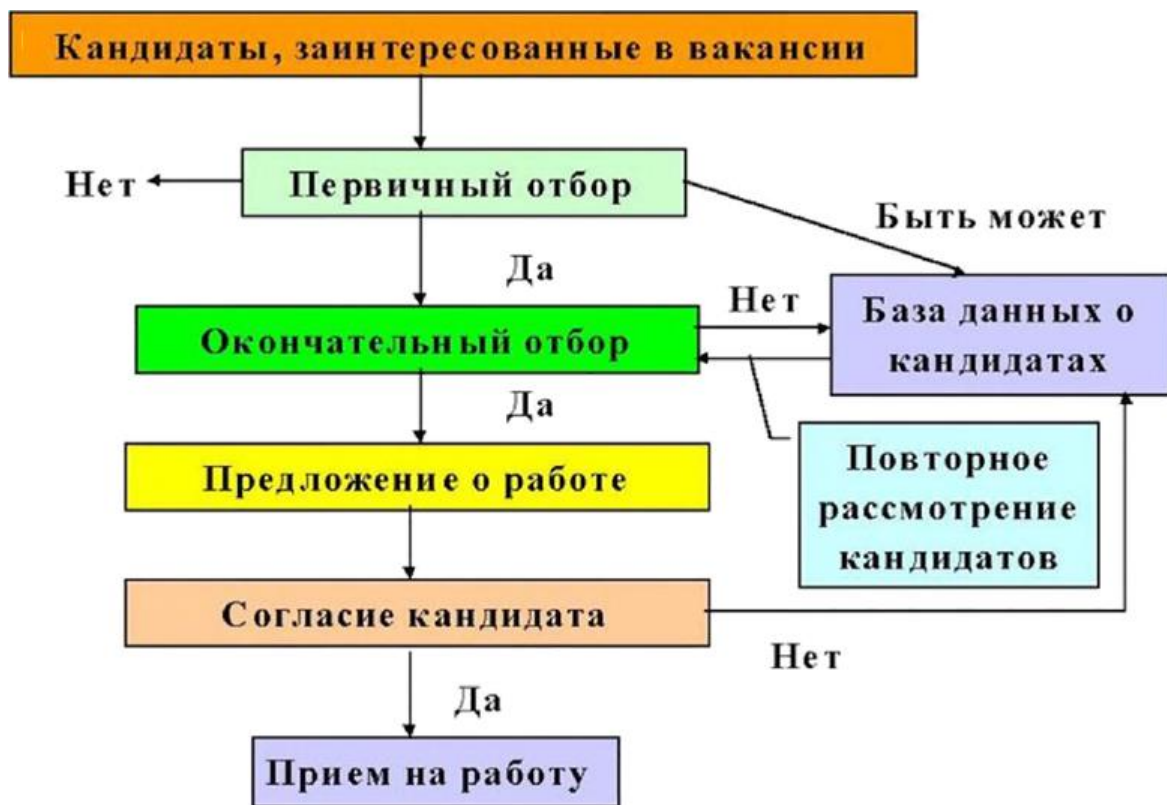
в) резюме арқылы жұмысқа қабылдау, бұл кезеңде резюмеге талдау лар жүргізіп, оның пайдалалығы жайлы шешімдер қабылдау жүргізіледі.

г) жеке сұхбаттар кезеңі. Бұл кезде жеке жұмысқы қабылдаушымен жауап алмасу жұмыстары жүргізіледі әрі бұл әдіс жеке адамдармен тікелей сұхбаттар арқылы көз жеткізіп қабылдауға мүмкіндіктер ашады.

д) тестілер өту кезеңі. Бұл қарапайым емтихан қабылдап алумен бірдей процесс. Жұмысқа қабылданушының қабілеттіліктерін мұқият тексеруге мүмкіндіктер береді.

е) жұмысқа қабылданушының жеке деректерін тексеру арқылы қабылдау кезеңі. Жұмыс өтелімін, істеген жерлерін және жинаған тәжірибелерін, заң алдындағы жауапкершіліктерін тексере отырып жұмысқа қабылдауға мүмкіндіктер береді.

Жалпы жағдайда барлық көрсетілген жұмысқа қабылдау кезеңдерінің тигізер пайдасы өте жақсы деп есептеледі. Соған қарамастан бұл кезеңдердің пайдаланылу реті, келесі персоналды іріктеу кезеңдері 2 суретте келтірілген.



2-сурет – Персоналды іріктеп таңдау кезеңдері

Бос вакантты орындарға үміткерлерді бағалайтын кадрлық қызметтің миссиясы, компания күткен нәтижеге қол жеткізе алатын қызметкерлерді іріктеп таңдау болып табылады. Шын мәнінде, қабылдау балдары – бұл ұйымдағы адам ресурстарының сапасын алдын ала бақылау нысандарының бірі болып табылады [18, 25 бет].

Жұмысқа қабылдау барлық вакантты орындарға үміткерлердің белгілі бір резервін қалыптастырудан тұрады, ұйым бұдан кейін, ол үшін ең қолайлы қызметкерлерді дайын деректер қорындағы ақпараттарға сүйене отырып іріктеп таңдауды жүргізуге қабілетті болады [4, 14 бет].

Әдетте, кандидаттарды іріктеп таңдау сыртқы және ішкі дереккөздер арқылы жүзеге асырылады. Сыртқы дереккөздерге: мемлекеттік және коммерциялық жұмыспен қамту агенттіктері, жоғарғы және кәсіби оқу орындарының тапсырыс берушілері, кәсіби газеттер және журналдар, арнайы сайттардағы берілген жарнамалар болуы мүмкін. Әдетте көптеген ұйымдарда ваканттық орындарды сатылық даму үшін өз қызметкерлеріне, шеттен адамдар іздегенен гөрі ұсынатындығы мәлім, ол әрі арзанға түседі, бірақ бұл кездері жаңа идеялары мен көзқарастары бар адамдардың ұйымдарға тигізер пайдасынан шет қалып кою зиянының тиюі мүмкін [15, 36 бет].

1.2 Жұмысқа қабылдау критерилері мен моделдері

Жұмысқа қабылдау критерилері жұмыскердің денсаулығын, тәжірибесін және жеке сипаттамаларын толық қамтитындай етіп қалыптастырылуы керек: әрбір критерии бойынша «Эталонды» талаптар деңгейі сол ұйымда бұрыннан жұмыс жасайтын, өздерінің міндеттерін толықтай меңгерген жұмыскерлердің сипаттамаларынан алып өңдейді [27, 44 бет].

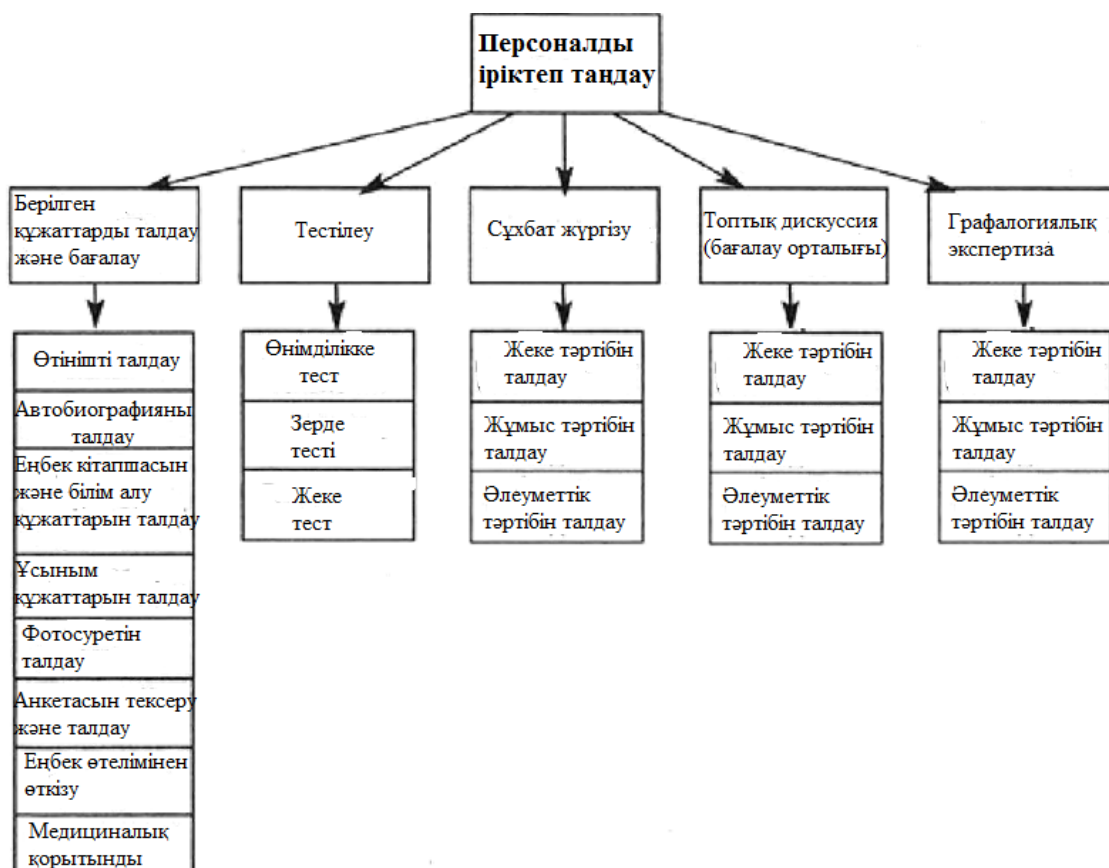
Көптеген ұйымдар жұмыскерлерді олардың алған білімдері бойынша қабылдайды. Егер олардың көрсеткіштері тең болса онда, білімі жоғарысын таңдайды. Бірақ бұл сипаттамалар жұмыстағы жетістіктерімен байланысты болып, білімділік критерилері орындаған жұмыстарының талаптарға сәйкестігімен өлшенуі міндетті болып табылады. Сол себепті жұмыс беруші алған білімдерінің мазмұнына көңіл бөлуі керек.

Тәжірибелік өтілі жұмысшы квалификациясын бейнелейтін маңызды критерилердің бірі болып табылады. Сондықтан көптеген жұмыс берушілер, жұмысқа қабылдау кезінде жұмыс өтілімдеріне көп көңіл бөледі.

Жұмыс өтелімін өлшеудің бірден-бір әдісі ол жұмысшылардың бір орында жұмыс жасау уақытына да байланысты. Еңбек өтелімі әртүрлі әдістермен өлшенеді: бір орында, бір лауазымда қанша уақыт жұмыс жасады, бір ұйымда қанша уақыт жұмыс жасады және т.с.с.

Көптеген жұмыс түрлері жұмысқа тұрушылардың физикалық жағдайларының дұрыстығын талап етеді. Осы мақсатта физикалық және медициналық сипаттамаларын да критерилер ретінде беріп кетіп жатады.

Мамандар (А.Я.Кибанов, В.А.Дятлов, А.Ваганов, В.К.Тарасов, И.Г.Чумарин, Н.В.Демин, Дж.Купер) персоналды жұмысқа қабылдау кезінде бес негізгі білімдерін тексеру аспаптарына көңіл бөлгенді дұрыс деп табады (3 сурет).



3-сурет – Білімдерін тексеру әдістері

Білімдерін тексеру әдістерінің дәстүрлі әдістеріне:

- алдын ала таңдаулар жүргізу;
- анкеталау;
- тестілеу;
- сұхбат алу;
- бағалау орталықтарынан өткізу;
- ұсыныстарын қарау және еңбек өтелімін тексеру.

1.3 Фармацевтикалық аймақтағы кадрлардың резервін қалыптастыру ерекшеліктері

Ваканттық қызметке кандидаттарды іріктеп таңдау кезінде ұйым қызметінің салалық спецификасына көңіл аударған дұрыс.

Қазіргі кездегі нарықтың дәріхана саласындағы жеке саудалар жүргізу кезіндегі ерекшеліктің бірі ол бұл саладағы кадрлардың жетіспеушілігі болып табылады. Бұл қазіргі кездегі дәріханалар санының жедел өсуімен, әрі ЖОО мен колледждердің шығаратын провизорлары мен фармацевтерінің жетіспеушілігімен байланысты, сонымен қатар бұл саладағы жас мамандардың төлем ақылары мен карералық өсулерінің болмауымен де байланысты болып

отыр. Міне сол себепті, дәріханалық сегменттер үшін «оқып үйретілетін» стратегияда жұмысшыларды қабылдау сипаты, және ол жұмыскерлердің сол ұйым негізінде квалификацияларын арттыру жақтарын қарастыру жағдайлары жасалады [17, 59 бет].

Осы себепте фармасевтика саласындағы кадрларды іріктеп таңдау жүйесін құру мәселесі қазіргі кезде маңызды талқыланатын тақырыптардың бірі болып табылып отыр. Сондықтан бұл салада жұмыскерлерді қабылдау кезінде тестілеу жүйесін қолдану көп қаржыны, көп уақытты, көп көңіл бөлуді керек етпейтін тестілеу жүйесі арқылы шешілуі керек деп шешіп, тестілер қалыптастыру жұмысын жүргізуді дұрыс көрдік. Яғни жұмысқа қабылдауды басшылардың қатысуыншысыз кез-келген дәріхана менеджері тексеру жұмыстарын жүргізіп, нәтижелерін электронды түрде басшыларға жіберіп, басшылардың қабылдаған шешімдерін жылдам, жұмысқа қабылдаушыларға ұсынып шешім қабылдауды жеделдете алады.

1.4 Отандық және шетелдік ұйымдардың тәжірибелеріне сүйеніп персоналды іріктеп таңдау

Дәріхана саласында квалификациялы мамандардың жетіспеушілігі кездеспейді. Провизорлар және фармасевтер көп жағдайда 1-2 жыл жұмыс істегеннен кейін профилдерін өзгертеді екен, тек 30-40 пайызы ғана осы мамандық бойынша жұмыс істеуге қалады екен. Олардың өзі осы саладағы бизнестерін жалғастырғандары болып табылады екен. Сол себепті отандық және шетелдік іріктеп таңдау тәжірибелерін зерттей келе, аса бұл жұмысшыларға қатаң талаптар қойылмайтындығын білдік. Бірақ білімдерін тексермей оларды жұмысқа алу өте дұрыс емес қабылданған шешім болып табылады. Сондықтан тестісіз немесе қандай да бір білімдерін тексеру әрекетінсіз жұмысқа қабылдау дұрыс емес деген шешімге келдік.

2 «Дәріхана» АҚ қызметтік жүйе орталығының талдануы

2.1 «Дәріхана» АҚ жалпы сипаттамасы

Дәріхана АҚ Алматы қаласы, Ауезов ауданы, Жетысу 3 шағын ауданы адресі бойынша қызмет атқаруда. Дәріхана қызметінде 4 адам жұмыс атқарады. Әрбір жұмысшы фармацевт техникалық талаптарға сай жұмыстарын атқарады. Қазіргі кезде дәріхана сату, сақтау және басқа қалаларға дәрі-дәрмектерді жіберу жұмыстарымен айналысады.

2.2 «Дәріхана» АҚ персоналды тестілеу жүйесінің тағайындалуы

Персоналды тестілеу жүйесі (ПТЖ) қызметкерлерді тестілеудің барлық процестерін автоматизациялауға арналған. Тестілеу қызметкерлердің білімдерін және біліктіліктерін дәріхана саласы бойынша орындауға мүмкіндіктер береді.

Жүйе келесі әрекеттерді автоматтандыруға арналған:

- тестілеуден өтетін қызметкерлердің тізімін функционалды міндеттері, қызметтері, штаттық кестелері және басқа да белгілері бойынша бөліп қалыптастыру;

- пайдаланушылардың функционалды ролдері, категориялары, қиындығына және басқа да белгілеріне байланысты бөліп тестілік сұрақтарды қалыптастыру;

- тестілеуді орындау уақытын бақылау;

- тестілеу нәтижесін қадағалау;

- тестілеу нәтижесінің талдауын, есепберулері мен диаграммаларын қалыптастыру;

- деректерді басқа негізгі жүйеге экспорттау немесе файлдар түрінде баспаға шығаруды қалыптастыру;

- тестілеуді орындау мүмкіндігі, әрі тексерушілерге арналған тестілік сұрақтардың жауаптарын тексерушілердің көре алу қабілеттіліктерін қалаптасыру.

2.3 Персоналды тестілеу жүйесінің қысқаша сипаттамасы

1) Жүйе Қазақстан Республикасының қазіргі кезде қолданыстағы еңбек жайлы (қолданыстағы еңбек заңнамасының және Қазақстан Республикасының заңының еңбек және азаматтық құқықтық қатынастарды есепке алу «Қазақстан Республикасындағы еңбек туралы» талаптарына) заңнамаларына толығымен сәйкес келеді.

2) Қызметкерлерді аттестациялауды орындау және бақылауды қамтамасыздандырады. Тапсырыс беруші программалық қамтаманы қолданғанға дейін қамтамасыздадыруы немесе өзі толықтыруы керек құжаттары:

- тестілеу, аттестациялау және бағалау әдістемесі және тестілік сұрақтар тізімі;

- дайын тестілік, аттестациялық және бағалау беттері. Тестілеуді, аттестациялауды және бағалауды жүргізу алгоритмі.

- тестілеудің, аттестациялаудың және бағалаудың нәтижелері бойынша ұсыныстарды қабылдау алгоритмі (жұмыстан шығару, қызметін түсіру немесе квалификациянды разрядтарын көтеру, қосымша оқуға жіберу).

Өңдеушілер тестілеу, аттестациялау және бағалау құралдарының аспаптық-программалық қамтамасыздандырылуын, жобалау жолдарын және программалық қамтамасын ұсынады.

Бұл жобаның басқа жобалардан артықшылығы тестілер базасын өзгертіп басқа да салаларда қолдануға мүмкіндік беретін модулінің болуы болып табылады. Керек болған жағдайда, жаңа тестілерді сақтау, қалыптастыру мүмкіндіктері де қарастырылған.

Динамикалық түрде тестілеу базасын қолданылатын, тестілеу программасын құру әрі тестілеуді корпоративті желіге қосылған компьютерлік кластарда немесе жұмыс орындарында тікелей өткізуді қамтамасыздандыратын мүмкіндіктері де қарастырылған, яғни:

- кез-келген филиалдан деректер қорына қатынай алу құқығының болуы (филиалдағы жүргізілетін тестілеу, аттестациялау және бағалау кезіндегі жұмыстарды тіркеу, баспаға шығару жұмыстарын жеңілдету мақсатында жасалған);

- тестілерді, бағалау беттерін, нәтижелік балл шығаруды орындайтын кешендік программа жасалғандығы;

- деректерді қайта енгізуді керек етпейтін кадрлық есепке алу модулінің болуы;

- қызметкерлердің корпоративтік желідегі адрестеріне тікелей ұсыныстарды, нәтижелерді бағыттай алу мүмкіндігінің болуы.

2.4 Программалық қамтаманы қолдану және функционалдау ортасының ерекшелігі

Жүйе MEDICAL.GUIDE ортасында өңделген және қазіргі кезде функционалдануда, әрі корпоративтік деректерді басқару және бүкіл әлем бойынша ынтымақтастықта жұмыс істеу жағынан жақсы платформа ретінде өзін танытты.

Программалық қамтама артықшылықтары:

– электронды пошта инфрақұрылымын, хабарлама жіберу және коммуникациялар жүйесін қолдауы;

– жоғары деңгейдегі қауіпсіздікпен қамтамасыздандыруы және ақпараттарды қорғай алу қабілеттілігінің болуы, деректерді шифрлау және электронды қолтаңбаны қолдануы, жүйедегі қолданушылардың әрекеттерінен қорғай алуы.

– интеграцияланатын программалық қамтама, яғни барлық ақпарат көздерімен соның ішінде, реляционды ДҚБЖ, өндірісті басқару ресурстарымен, Internet ортасымен және web сервистермен интеграциялануы.

– сенімділігі қосымша сенімділікпен қамтамасыздандыру механизмдерімен қамтамасыздандырылған, сондай-ақ резервті көшіру және жүйені қайта қалпына келтіру жақтары да қарастырылған.

– иілгіштігі барлық танымал операциялық жүйелерде жұмыс жасай алуымен ерекшеленеді.

– администрациялау ыңғайлылығы және қарапайымдылығы жұмыстың автоматтандырылғандығымен шешілген.

2.5 Есептің қойылымы және қолдану аймағы

Есептің қойылымы және жүйені өндіріске енгізу келесі кезеңдер бойынша орындалады:

1) тапсырыс берушінің деректерін жинау және талдау (талаптары, құжаттар формалары, анықтамалық құжаттар шаблондары, еңбек келесім шарттары, талап етілетін есепберу құжаттары, бизнес-процесстері, аттестационды, бағалау және қызметкерді тестілеу тестері);

2) жүйені ашып қарау және оны тапсырыс берушінің талаптары мен бизнес-процесінің жүргізілуіне сәйкес адаптациялап программалық қамтама жасап шығару;

3) пайдаланушыларды оқып-үйрету және программалық қамтаманы өндіріске енгізуді орындау;

4) жүйені жұмыс орнында тестілеу және корректировкалар енгізу және нәтижелері бойынша жүріп өту жұмыстарын орындау;

5) өндіріске толығымен енгізу;

6) кепілдік бере отырып техникалық қолдау.

3 Персоналды тестілеу жүйесін жобалау және программалық қамтамасын өңдеу

3.1 Программалық қамтаманы жобалау

Объектіге-бағытталған программалық автоматтандырылған жүйелерді жобалау әрі талдау моделін құруға арналған унифицирленген моделдеу тілі қолданады. 1989–1997 жылдары іске асырылып қолданыла бастаған сәйкестендіру арқылы, қазіргі кезде барлық жерлерде UML моделдеудің унифицирленген тілін ортақ тіл ретінде қолданады.

Модель – бұл кез келген жүйе құрушының көзқарасы жағынан бекітілуі керек нақты әрекеттерді болжау немесе нақты орындалуы керек операцияларды болжау ұсыныстары. Модель қойылған міндеттерді немесе мәселелерді жылдам шешуге, нәтижелерін тиянақты түрде алдыға қойып алуға мүмкіндіктер береді. UML диаграммаларының әртүрін қолдану, әртүрлі актерлармен берілген міндеттерді немесе мәселелерді қарап шығуға әрі, сондай ақ бір қозғалыста және бәрдей контексте орындалатын өзара байланыстарды анықтауға жол ашады.

UML (Unified Modeling Language) объектіге-бағытталған программалық жүйелердің моделін жобалау, талдау әрі эксплуатацияға енгізіп жүзеге асырғанға дейінгі кезеңдерді қамтуға, моделдерін құруға арналған стандартты унифицирленген моделдеу тілі. UMLде программалық жобалардың нәтижелерін құжаттауға, құрастыруға, спецификациялауға әрі визуализациялауға арналған элементтерді қолдануға болады. Жалпы алғанда UML – бұл программалау тілі емес, ол моделдеу тілі, бірақ оның қолданған моделдері Java, Visual Basic, C++, Object Pascal, Ada 95 программалау тілдеріндегі текстерді және реляционды деректер қорының кестелерін тікелей өз моделдеу аймағында тасымалдауды орындайды.

UML унифицирленген моделдеу тілі, моделдердің құрылуына арналған жалпы қабылданған стандарт болып табылады. Бұл тілдің спецификацияларында тілдің моделдерінің түрлерінен және олармен жасалатын ережелер жазылған. Программалық қамтама жүйелерін өңдеушілер арасындағы қолдануы біртұтас стандарттар, барлық қолданушылардың бір тілде қатынасып әрі түсінісіп жатуларына, моделдің құрушылары қалағандау әрекеттерді орындайтындай етіп жобалауларына мүмкіндіктер береді. Сондай ақ жабаланып, құрылған моделдердің қолданудағы бірімәнділігін сақтайтын программалық қамтама кодтарының моделдерінің негізінде RationalRose немесе RationalXDE және басқа да осы сериядағы арнайы программалық қамтамаларды қолдануғы мүмкіндіктер бере алады.

UMLдің сөздігі арнайы осы тілдің үш түрлі құрылыс блоктарынан тұрады: мәндер, мәндер арасындағы қатынастар және берлгілі бір тәртіптерге сүйеніп осы мәндерді берлігі біл байланыстармен қосуға мүмкіндік беретін диаграммалар.

Программалық жүйелердегі мәндер аймағын модельдеу үдерісі нақты қазіргі дүниедегі абстракциялар болуы мүмкін. Абстракциялардың негізгі міндеті ақпараттар аймағын құру бағытындаға қайта программаларда қолдануға мүмкіндік беретін негізгі әрекеттер болып табылады. Мәндер аймағының моделін тұрғызу кезінде нәтижелер ала білу әдістемесі қолданылады. Бұл кезде сақталу керек бірден бір қағида, ол жүйедегі бастау алатын объектілердің кілттерін сақтай білуінде. Одан кейін, сыртқы моделдің қандай объектілермен қарым-қатынаста болатындығын ажыратып, тізбекті орындап үйренуінде. Көрсетілген әдіспен орындалатын объектілер арасындағы қозғалыстарды көрсетуге немесе жүйенің динамикалық сыртқы объектілерден ішкі объектілерге қозғалысы қалай жүретіндігін, әрі статикалық моделдің нәтижесінде іштен сыртқа жүретін әрекеттерін анықтап отыруға көмегін тигізеді.

Мән аймағын моделдеу көрсетілген әдістердің статикалық түрі болып табылады. Мән аймағының моделінің құрылымын тұрғызу абстракцияны ерекшелеуден басталады. Бұл нақты өмірдегі концептуалдық объектілерде, әрі үлкен жүйелерде кездеседі. Объектілерді моделдеп, жобалау кезінде сол жүйеге бағытталған ақпараттармен қамтамасыз ететін құрылымдық әдістерді қолдану талап етіледі, себебі берілген бір ортада құрылған тапсырмаларда сол объектілердің өздері болуы керек. Ақпараттарға берілетін талап нақты өмірге карағанда жылдам өзгереді, сондықтан барлық талаптар сақталуы керек. Объектілердің моделінің негізгі әрі статикалық моделдерінің мән аймақтарын тұрғызу сол абстракциялардың моделдерін шақырумен жүзеге асырылады. Оларды шақыру, шығару және әрекеттерді сипаттау объектілерімен, класстарды сипаттауда қолданылады.

Программалық қамтамаларды жобалау әдістері мен программалық қамтамалардың өмірлік циклдарымен қамтамасыздандырады.

Унифицирлен моделдеу тілінің баста міндеті командалық және жеке жобалау жүргізуге әрі оған керекті аспаптарды жеңіл түрде қолдануға мүмкіндік бере алуында болып табылады. Бұл тіл сонымен қатар текстуальды және визуалды жобалаудың технологиялары мен әдістемелерін өз аспаптары ретінде қолдана алады. Визуалды модельдеу (visual modeling) – бұл нақты өмірдегі объектілер мен ұғымдарды көрсете алатын әрі өзі сол объектілерден көріністер таба алатын абстракциялар арқылы мәселелерді қабылдау әдісі деп түсінсек болады. Модельдеу мәселелерін талдау, қадағалау, бақылау барлық сол салада қызығушылық танытатын актерлар (қолданушылар, пайдаланушылар, пәндік аумақтағы арнайы мамандар, талдаушылар, дизайнерлер, админдер және т.б.) арасында ақпараттар алмасуын, программалық қамтамалар мен деректер қорын жобалау, әрі құжаттарын дайындауды қамтамасыздандыратын пайдалы құрал болып табылады. Моделдеу талаптарын орынымен қабылдауға, жүйенің дизайнының сапасын дамытуға, оның басқару деңгейін жоғарылатуға өз себебін тигізеді. UML – объектіге-бағытталған арнайы жүйелермен құрастырылатын, мәндердің құжаттарын белгілеу және көріністерін жақсартуда қолданылатын тіл.

RationalUnifiedProcess – қазіргі кезде еңкөп кездесетін, әрі қолданылатын әдістемелердің бірі. Олар RationalSoftware компаниясымен өз өнімдерін пайдаланушылар арасында қолдау үшін өңделген тіл, олардың саны оннан көп деп есептеледі (солардың ішінде ең көп кездесетіндері – RationalRose және RequisitePro болып табылады). RationalUnifiedProcess үш атақты адамдармен құрылған – олар Гради Буч, Ивар Якобсон әрі Джеймс Рамбо (Rumbaugh). Олар қиын программалық қамтамаларды өңдеуде жүйелерін жасауда үлкен жетістіктерге жеткен атақты адамдар деп есептеледі, оны RUP бейнесінен-ақ көз жеткізуге болады.

3.2 Rational Rose аспабы

Программалық қамтамаларды құру әдістемелері белгілі бір аспаптарсыз болмайды. Қазіргі кездегі нарықта аспаптардың ауқымды түрлерінің бар екендігін қарапайым жобалардан бастап, объектілерді моделдеудің кездесетін ең күрделі жүйелеріне дейін қарастыруға болады. Біз өз дипломдық жобаларымызды моделдеу барысында RationalRose аспабына сүйене отырып жасадық.

RationalRose аспабының өнімдерінің сериялары құрастырушыны нақты жүйелерде және клиент-сервер жүйелерінде қолдануға болатындай және қазіргі уақыттағы бизнес талаптарын қанағаттандыратындай тиімді әрі сенімді шешімдер қабылдауға қабілетті ететіндей, көмекші визуалдық моделдеу құралдарының жиынтығымен қамтамасыз еткен. RationalRose аспаптары бірегей стандарттарға негізделіп жасалған және оларға жақын салалардағы бизнес-үдерістерді тиімділеу талаптарын орындайтын, компьютермен онша танысемес актерлармен қатар, программалық қамтамалардың логикасын моделдеу аспаптарын қажет ететін мамандарға да қолдануға оңай.

RationalRose – UMLдің әдістемелерін тарату аспабы. Бұл аспабқа ұқсас міндеттерді шешуде қолданылатын басқа да визуалды аспаптар жеткілікті. Бірақ ондай программаларды ақшаға сатып алу керек болады, сол себепті біз өз жобамызды жасауда қол жетімді аспалты қолдануды жөн көрдік. Өңдеу функцияналдығын сипаттау үшін қолдану варианттары (прецеденттер) диаграммасы (онда талаптар және шектеулер әрекеттер арқылы көрсетіледі), функциялардың орындалу реттері әрекеттердің объектілердегі тізбектер диаграммасы (объектілер арасындағы хабарламалар алмасу), өңдеу элементерінің қарым қатынасы және барболуының сипаттамасы кооперация диаграммалары (уақытқа тәуелділігі жоқ) қолданылады. Интерфейстерін жобалау объектілік жобаның өңделуімен орындалады, Қолдану варианттарының диаграммасы немесе прецеденттер диаграммасы барлық қалған жобадағы диграммалар үшін арналып жасалады. Прецеден сөзі, негізінен, оның жүйедегі атқаратын роліне байланысты деп айтуға болады себебі, пайдаланушы жүйенің қолдану варианттарының диаграммалары

(актерлар, әрекет орындаушы объектілер) арқылы беріледі. Сонымен, бұл диаграмма негізінен жүйедегі орындалатын немесе орындалуы керек әрекеттерді сипаттау үшін жасалады, әрі: кім, қашан және неістеуі керек екендігін білу үшін орындалады. Келесі диаграмма бұл тізбек диаграммасы ол жеке қолдану вариантындағы әрекеттерді объектілер арқылы орындауды түсіндіреді, әрі бұл диаграммада актер әрекеттерді орындау кезінде кіммен (немен) қарым қатынаста болуы керек екендігін көрсетеді. Сондай ақ бұл диаграммада қолдану варианттарының орындау уақыты анықталады, ол қолданылатын жүйеге нақты уақыт бойынша қалай, қандай қызметтер орындаулары керек екендігін көрсетеді әрі жүйе, ON-LINE түрінде жұмыс жасай алатындығын ерекшелеп береді. Сонымен қорыта айтар болсақ бұл диаграмма қолданушы варианттары арасындағы объектілер арасындағы өтулері мен хабарламалар алмасуын анықтайды. Ескерту бұл көрсетілген диаграммада қолданушылар немесе пайдаланушылар ретінде интерфейсте қолданылып отырған элементтерді және басқа да ақпараттық объектілерді алуға болады.

Объектілер нақты немесе абстракттілі мән ретінде анықталады. Объект дегеніміз программалық қамтамада нақты бір шектеулер, ұғымдар және мәндермен көрсетілетін түсінік немесе абстракция.

Жасалатын жүйенің әрбір объектісінің міндетті анықталатын үш сипаттамасы болады, олар объект жағдайы, объект тәртібі және объектінің біркелкілік белгісі.

Объект жағдайы бұл атрибуттардың қасиеттерінің жиынтығы, әрі басқа абстракциялармен қарым қатынас арқылы анықталады. Объектілердің тәртіптерінің сипаттамасы, объектілердің функционалдық өмірлік циклдарын қамтуы, басқа объектілер сұранысына сәйкес әсерлерін зерттеп, олардың операцияларының жиынтығы түріндегі іске асырылуы деп түсінуге болады. Біркелкілік немесе бірмәнділік белгісі объектінің басқалармен үнемі әмбебаптығын анықтайды, олар басқа объектілермен сәйкес келген жағдайда класс объектілерінің тобын ортақ атрибуттары, ортақ қасиеттері, ортақ тәртіптері немесе функционалдары, ортақ семантикасы және басқа объектілермен байланыстары арқылы анықтауға мүмкіндік береді. Класты басқа объектілердің түрін құруға арналған тәртіп немесе шаблон деп айтуға болады. Әрбір жүйедегі объект тек біркластың данасы болып табылады. Класты құрған кезде оны міндетті түрде құжаттандыру керек. Себебі бұл сипаттама класс құрылымынемес, оның мәнін беру үшін керек болады.

RationalRose өнімдерінің сериялары құрушылардың нақты уақыт жүйелерінде және клиент-сервер жүйелерінде қолдануға жарайтын, әрі қазіргі уақытта бизнес талаптарын барынша қанағаттандыратын тиімдіде, сенімді шешімдерді қабылдауға көмегін тигізетін визуалды моделдеудің толық барлық құралдарының жиынтығымен қамтамасыздандырады.

RationalRose аспаптары бірегей стандарттарға сәйкес негізделген жүйе. Олар модельдеуді, оларға жақын салалардағы бизнес-үдерістерді тиімділеуге арналған, компьютерлермен танысемес пайдаланушылардың қатарына жататын, программалық қамтамалардың логикасын моделдеу аспаптарын қажет

ететін мамандар мен қолданышалраға оңай дүние болып абылады.

3.3 Прецеденттер диаграммасы

Программалық қамтаманың негізгі сатысы ол прецеденттер диаграммасы болып табылады. Прецеденттер диаграммасы – бұл арнайы болжанған жүйені жүргізудің құжаттық моделі.

Прецеденттер диаграммасындағы кездесетін элементтер, олар: прецеденттер немесе әрекеттер, актерлер және олардың арасындағы қатынастары. Прецеденттер диаграммаларының көмегімен жүйеге арналған прецеденттердің статикалық көрінісі жасалады.

Прецеденттер диаграммасы қолданушының көзқарасы жағынан жүйенің барлық орындалуы керек тәртібін анықтайды. Прецедент диаграммасы жүйенің динамикасын бастапқы моделдеуге арналған маңызды аспап ретінде қарастырылады, жобаланатын жүйелерге бастапқы талаптарды беруге арналған, кейінгі ол өндеулердің нәтижелерін көруге мүмкіндік беретін формаға әкелетін барлық жалпы жүйе талаптарын тіркеуді орындайды. Барлық қолданылатын диаграммаларда соның ішінде прецедент диаграммаларында да ескертулер мен шектеулерді қосып беріп отыруға болады. Содан басқа прецедент диаграммалары жүйедегі өзінің үлкен фрагменттерін бейнелеуі үшін топтастыру моделдерді, яғни пакеттерді қолданады.

Актер – бұл диаграммадағы нақты әрекеттерді орындаушы элемент. Ол тікелей өзара әрекеттесетін жүйенің сыртындағы объектілердің ролі.

Прецедент элементі – бұл жобаланатын жүйемен орындалуы керек және басқа актерлар үшін көрінетін нәтижені бере алатын әрекеттердің кезектілігінің сипаттамасы.

Актерлар жобаланатын жүйе жұмысына қажет етілетін сыртқы дүниенің сипатын көрсетеді, ал прецедент элементтері актер құқығы бойынша жүйемен орындалатын әрекеттердің тізбегін көрсетеді. Сондай ақ жүйеде активті субъектілер және жобаланатын жүйенің атқаратын қызмет түрлеріне сәйкес олардың арасында қарым қатынасты орнату керек. Ол қарым қатынастар ассоциациялық қатынас түрімен беріледі.

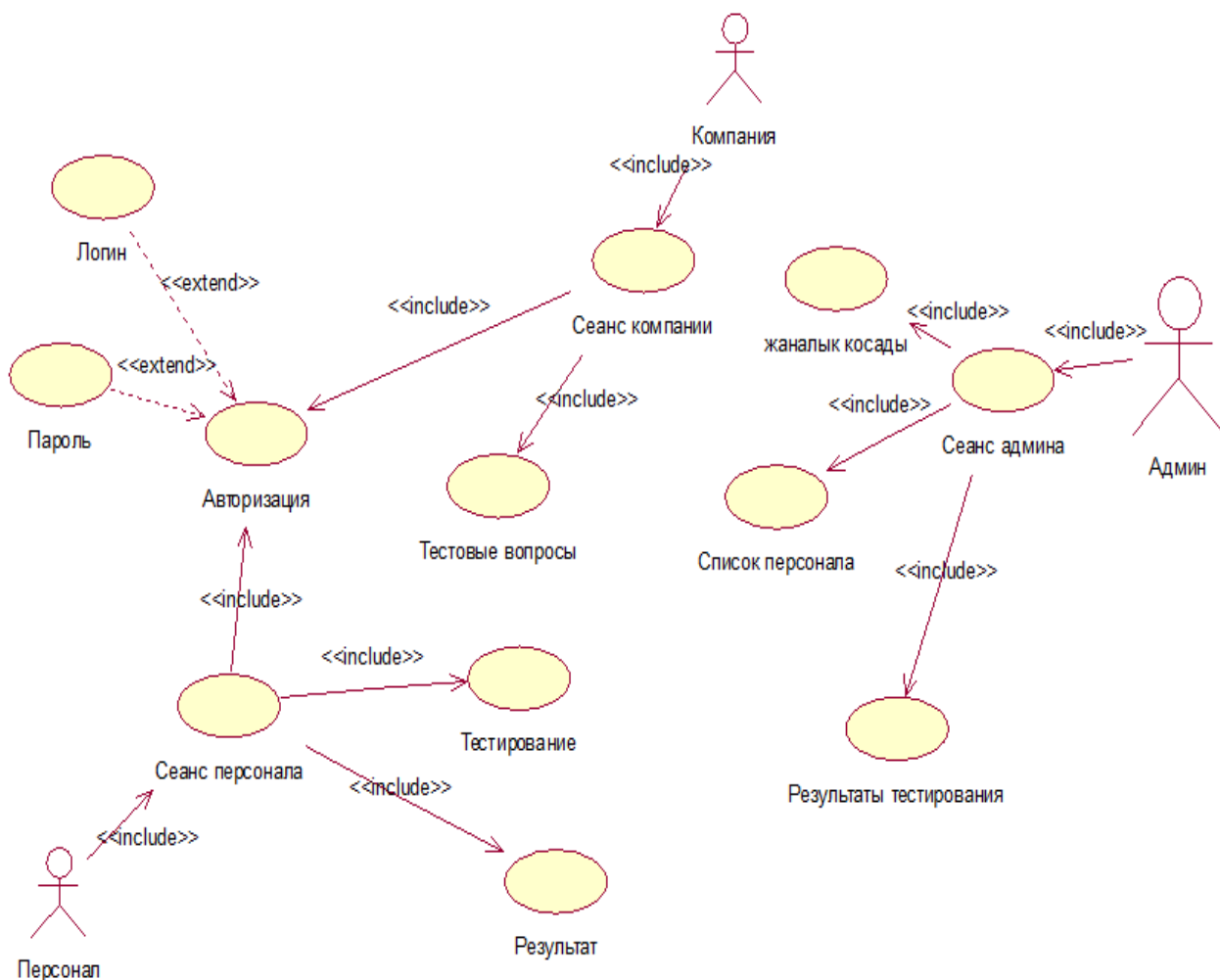
Прецеденттер арасындағы екі типтегі тәуелділік қатынасымен беруге болады:

- Include – жүйемен қосылатын қатынастар;
- Extend – жүйемен кеңейтілетін қатынастар.

Қосылатын қатынастар актермен қосылатын функцияның орындалуымен бірге прецеденттің сол функциялармен анықталғандығын әрі осы прецедентке басқа да жүйедегі прецеденттердің біріге алатындығын көрсететін қатынас.

Кеңейтілген қатынастар актермен шартты емес жағдайда туындауы мүмкін, міндетті емес функциялардың жиынтығын көрсететін қатынас. Прецедент элементі әрбір актердің құқығына сәйкес жүйеде орындалатын әрі

актерлар үшін нақты бір нәтижелер бере алатын әрекеттер тізбекгін немесе бірнеше тізбектемелерін сипаттау элементі. 4-суретте прецеденттер диаграммасы келтірілген.



4-сурет – RationalRose ортасында құрылған прецеденттер диаграммасы

Қолдану вариантының диаграммасы немесе Use Case Diagram, бұл жүйедегі активті субъектілердің әртүрлі графикалық көрсетілімін және оның қолдануын, сол немесе басқа жүйе варианттарының амалдарының арасындағы әрекеттерін көрсететін диаграмма. Жүйені жобалауды жоспарлағанда жүйенің кілттік функциялары және әртүрлі пайдаланушыларды немесе актерларды ұсынатын негізгі диаграмма, яғни Main Use Case Diagram құрылады.

3.4 Тізбектер диаграммасы

Өзара жобажағы объектілердің және актерлардың әрекеттесуінің диаграммасы, оның екі түрі бар, олар: тізбектер диаграммасы мен кооперация диаграммасы. Тізбектер диаграммасы, бұл объектілер арасындағы

хабарламаларды уақытқа сәйкес тәртіпке келтірудің айқындалған әрекеттесуінің диаграммасы ретінде қарауға болады, яғни бір уақыт аралығындағы іс-әрекеттерді жобалайды. Кооперация диаграммасы, бұл объектілер арасындағы хабарламаларды уақытқа тәйуелсіз қайда жіберетін, әрі кім қабылдайтын анықтайтын объектілердің құрылымдық ұйымдасуын айқындап беретін әрекеттесу диаграммасы десек болады. Өзара объектілердің және актерлардың арасындағы әрекеттесу диаграммаларының элементтері, олар объектілер, қатынастар, хабарламалар. Яғни өзара әрекеттесу диаграммасының басты міндеті ол объектілер және олардың арасындағы қатынастар арқылы, сонымен қатар объектілер немесе актерлар арасындағы бір-біріне жіберілетін хабарламаларды бір бірен қосатын өзара әрекеттесу түрлерін береді. Объектілердің өзара әрекеттесу диаграммалары негізгі жүйенің динамикалық көрінісін беруді қамтамасыздандырады.

Тізбектер диаграммалары жүйедегі орындалатын әрекеттердің тәртіптік сценарилерін бере отырып, хабарламалардың беру тәртібінің мағыналы айқын көрінісімен қамтамасыздандыруды орындайды. Әрі ол автоматты түрде алуға болатын кооперация диаграммасындағы көрінетін бөліктерін береді.

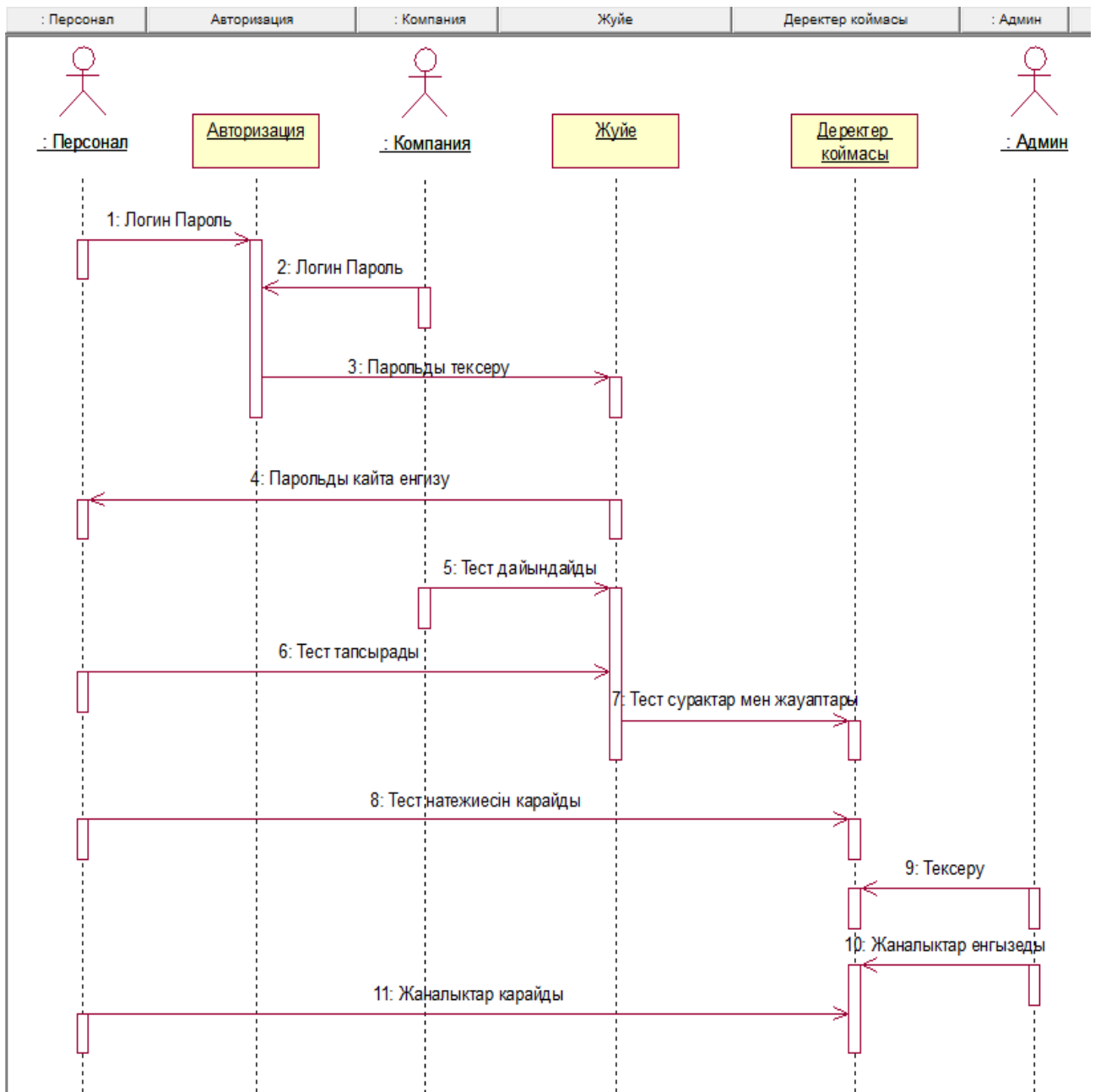
Тізбектер диаграммасы төрт негізгі элементтен тұрады:

– прецеденттердегі өзінен кейін орындалатын текстің әрекеттері. Ол солдан бастап, жоғарыдан төменге қарай жазылады. Ол терезедегі әрекеттерді сипаттайды, әрі жұмыс уақытындағы орындалуы ақпараттарды пайдаланады;

– "объект-класс" форматында объектілердің аты немесе объектлердің данасы нөмері, әрі класс объектісінің атауы жазылады;

– хабарландыру, бағыты бар бір объектіден басқа бағытталған келесі объектілерге берілетін әрекеттер туралы ақпараттардан тұрады. Келісілген уақытта орындалуы керек және осы әрекеттегі жүйенің нәтижелік жауабынан тұрады.

– операциялар. Тіктөртбұрыш түрінде беріледі, әрі үздік сызықпен жалғасады. Яғни, әдістерге сәйкес объектілерден тұрады. Тіктөртбұрыш ұзындығы олан кейнгі орындалатын басқару фокусына тәуелді, тіктөртбұрыш бітетін нүктеде басқаруды басқа объект иеленеді. Бұл өмір циклы деп аталады. 5–суретте жүйенің тізбектер диаграммасы келтірілген.



5-сурет – Rational Rose ортасында құрылған тізбек диаграммасы

Тізбектер диаграммасы алдында айтқанымыздай уақыт бойынша объектілердің өзара бірлесіп хабарлама алмасу операцияларының орындалу ретін бейнелейді және сценарийлерде қарастырылған функциялардың орындалуындағы үдерістерді арасында хабарламалар алмасатын объектілердің тізбектерін реттеуге кіретін объектілер мен класстар қатынасын береді. Тізбектер диаграммалары жалпы түрде LogicalView пакетінде прецеденттер диаграммасымен бірге ассоциацияланады.

UMLдің талаптарына сәйкес объектілердің тізбектер диаграммасында тіктөртбұрыш түрінде көрсетіледі де, ол объектілер диаграммасы деп аталады. Олардың асты сызылып көрсетіледі. Жоғарыда көріп отырғандарыңыздай объектілерді үш түрлі әдіспен айтуға болады: тек объектінің атауын көрсету

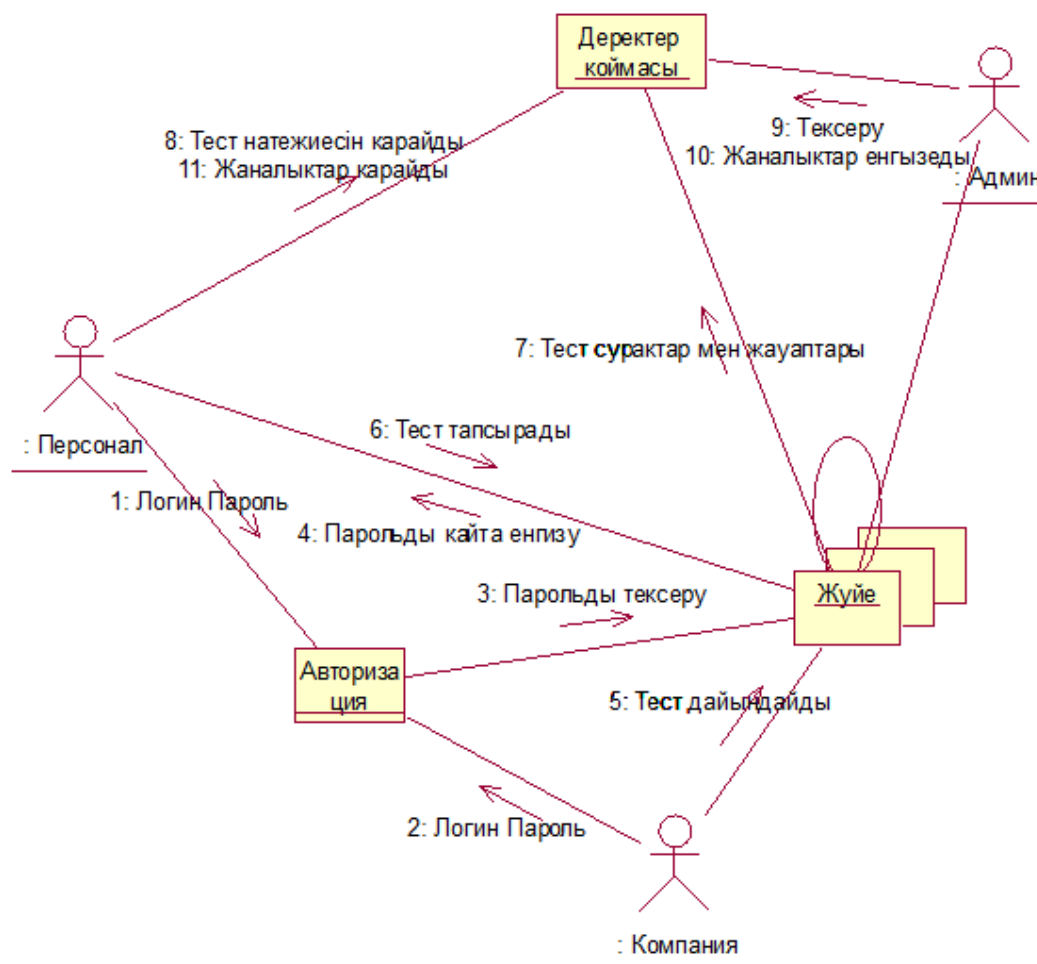
арқылы, объект пен класстың атауын көрсеті арқылы, әйтпесе класстың атауын шектеп көрсету арқылы (жасырын объекттер үшін).

Тізбектер диаграммасында активті субъектілер беріледі, олар өзара бірлесу фактілерімен жүйенің шекаралар класын қосады. Мұндай талаптар талдау үшін ерте кезеңдерде бекітілуі және интерфейстер талаптарының жеткілікті деңгейде құжатталуын талап етеді. Сол себепті активті субъектілерді нақты түрде шекараларын белгілеп, хабарламаларын реттеп, орындалатын операцияларындағы әдістерін көрсетіп, әрі ол жайындағы ақпараттарды дұрыс сақтап алған жөн. Ал мұндай ерекшеліктерді ұстану жобалаудың соңғы кезеңінде жүзеге асырылатын нәтижелерді дұрыс алуға мүмкіндіктер береді. Сондықтан, жобаланатын жүйенің даму барысына орай және осындай кішігірім кездейсоқтықтарды болдырмау үшін барлық шарттарды және талаптарды ұстанған дұрыс деп есептеледі.

3.5 Кооперация диаграммасы

Диаграмманың бұл түрі алдында айтып өткеніміздей автоматты түрде тізбек диаграммасынан немесе керісінше түрде орындалатын үлгідегі диаграмма болып табылады. Диаграмманың бұл түрі жалпы түрде берілген объектілердің барлық қабылданатын және берілетін хабарларын үлгілейтін және осы хабарламалардың үлгілерін көрсететін диаграмма.

Sequence және Collaboration диаграммалары жүйеде бір үдеріске әртүрлі көзқарас болғандығын қамтамасыздандырады, RationalRose Sequence диаграммасынан Collaboration диаграммасын құруға немесе керісінше орындауды береді, сондай ақ, бұл диаграммалар автоматты түрде бір біріне синхронизацияланады. 6–суретте кооперация диаграммасы келтірілген.



6-сурет – Rational Rose ортасында құрылған кооперация диаграммасы

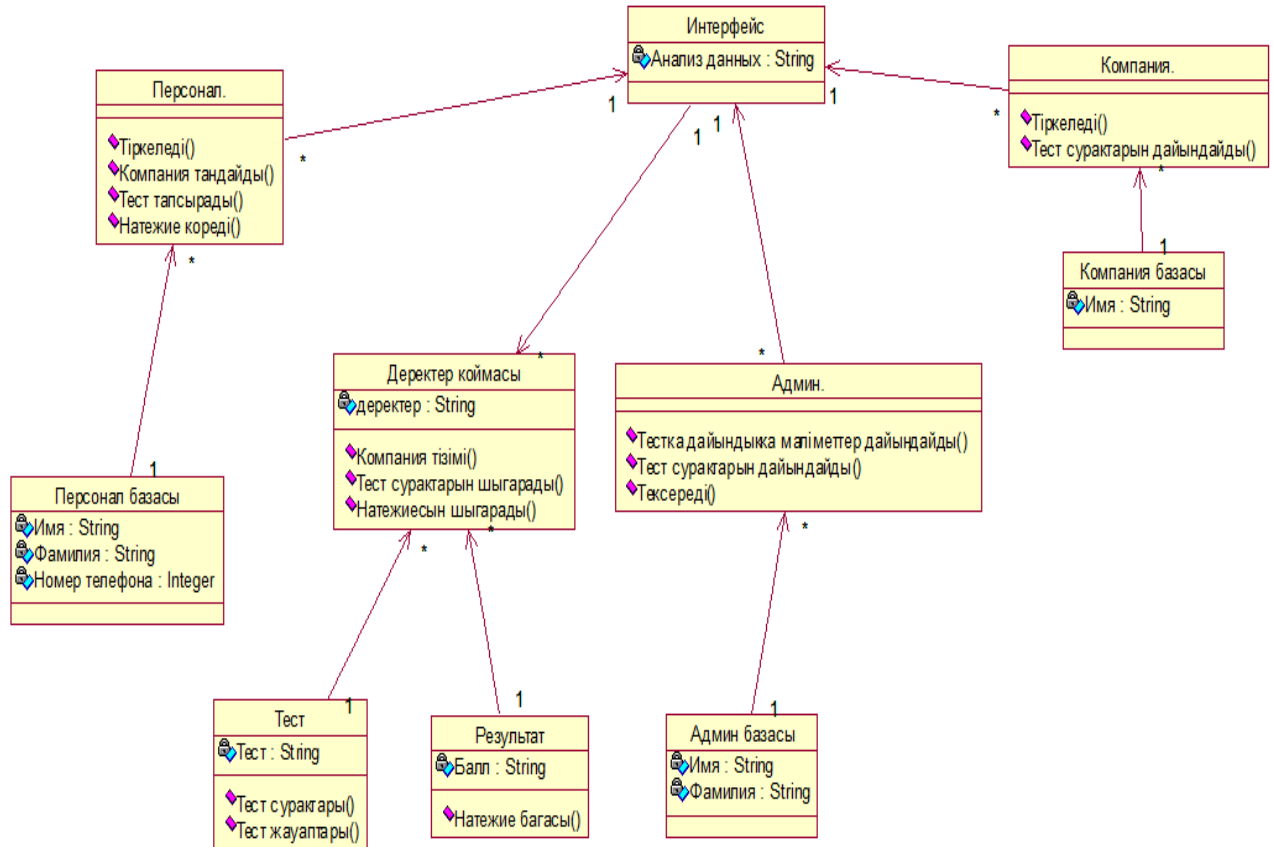
RationalRose ортасында кооперация диаграммасын жүзеге асырудың бірнеше әдісі қолдануға болады:

- стандартты RationalRose аспаптар суретіндегі әрекеттесу диаграммасын басу арқылы, әрі кооперация диаграммасын құру батырмасын таңдау;
- негізгі менюден келесі операцияны орындау: Browse → Interaction Diagram (Браузер → әрекеттесу диаграммасы) және новый кооперация диаграммасын құруды таңдау;
- контексті менюден келесі операцияларды орындау: логикалық көріністі, әрі жобаның браузеріндегі қолдану варианттарын көрсету үшін New → Collaboration Diagram (Новый → Кооперация диаграммасы) таңдау арқылы.

3.6 Кластар диаграммасы

Класстар диаграммалары жобандағы қолданылатын негізгі класстар, интерфейстер, кооперация әрі олардың қарым қатынастарын көрсетеді. Объектіге-бағытталған жобалау жүйелерін модельдеу кезінде, әдетте кластар

диаграммалары көп қолданылады. Кластар диаграммаларының мақсаты жүйенің статикалық көрінісін қамтамасыздандыру болып табылады. Объект және класс түсініктері бір бірімен тығыз байланыста. Класс, бұл объектілердің негізгі маңызды сипаттамалары. Класс диаграммасы моделдің басты логикалық көрінісіне береді, кластар диаграммасы 7-суретте келтірілген.



7-сурет – Rational Rose ортасында құрылған класс диаграммасы

Класс диаграммасы негізінен қосымша программалық қамтама кодын енгізу үшін маңызды диаграмма болып табылады. Кластар диаграммасының көмегімен ішкі жүйе шығарылады, яғни ол мұрагерлеуді сипаттауға және қосымша кластармен байланыс орнатуға мүмкіндіктер береді.

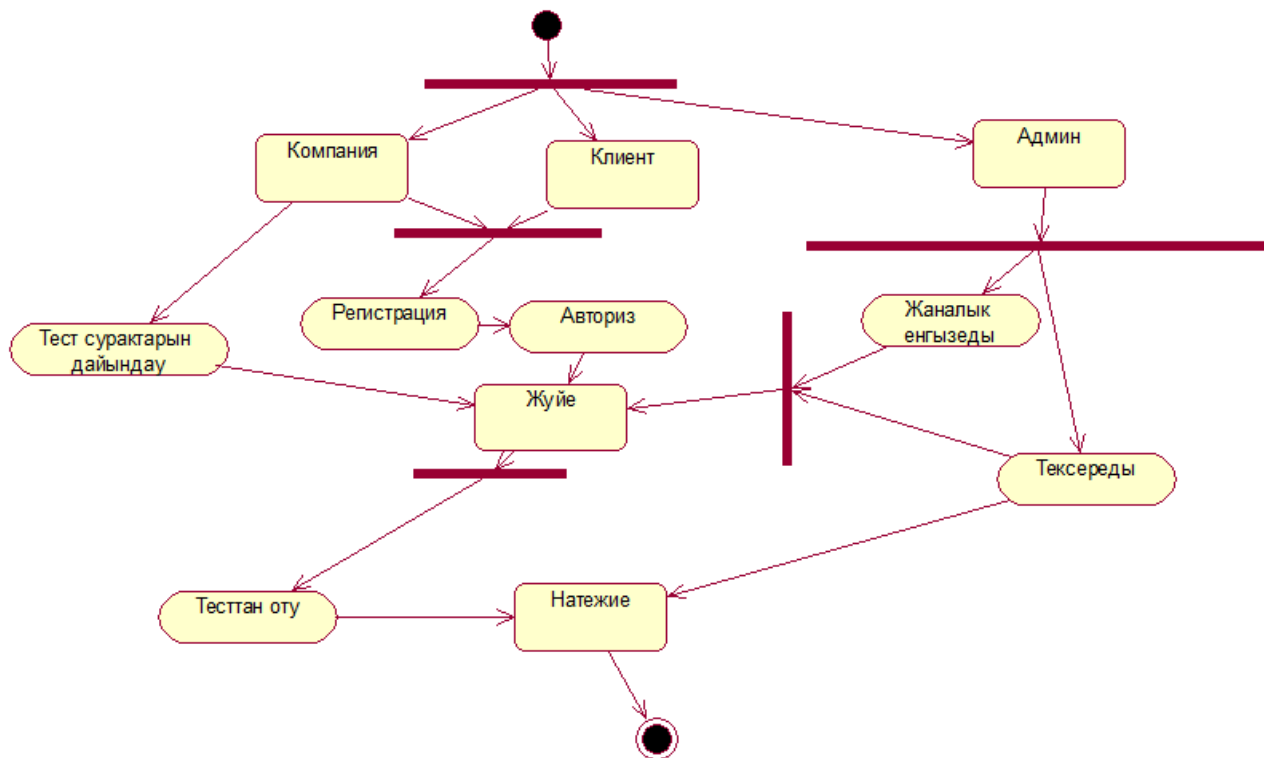
3.7 Күй диаграммасы

Күй (State) диаграммасы, бұл кейбір жүйеде қолданылған объектілерді орындау кезіндегі белгілі бір әрекет немесе оқиғаның орындалуын күтуде жүзеге асыратын тәртіптер жиынымен шартталады. Объектілердің күйі класы бір немесе бірнеше атрибуттарымен сипатталу мүмкін.

Белгілі тәртіптерге ие болатын жүйенің әрбір объектісі объектілердің тәртіптерінің сценарилерін жүзеге асыру үдерісінде белгілі бір әрекет жасау

арқылы бір күйден басқа екінші күйге өтуі мүмкін.

Күй диаграммасы (Statechart) тәртіптердің күрделі моделдерінен тұратын жүйенің объектілерінің күйлерін көрсетуге бағытталған. Бұл бағыттар бір меню пунктінен жүзеге асырылатын StateMachine диаграммаларының бірі болып табылады. 8-суретте күй диаграммасы келтірілген.



8-сурет – Rational Rose ортасында құрылған күй диаграммасы

3.8 Программалық қамтама өңдеу

Деректер қорын жобалап, одан кейін қосымша жасау керек, яғни аптека аймағындағы тестілеу аттестациялау және бағалау жүйесін жүргізуге мүмкіндік беретін, әрі әрбір пайдаланушылармен қарым-қатынас жүргізуге мүмкіндік беретін орта құрылады. Қалай құрылатыныды жайында және қандай талаптар мен міндеттер шелуі керек екендігін жоғарыдағы бөлімдерде айтып өттік, әрі диплом алдыңғы тәжірибе кезінде техникалық тапсырмасын жасап өткізгенбіз.

3.8.1 Интерфейсін өңдеу

Программаның интерфейсін өңдеу кезінде келесі үш келесімді қолдайтындай етеп жасау қажет:

1) пайдаланушыларға тікелей бағытталған, әрі соңғы қолданушының яғни персоналдардың қажеттіліктерін қанағаттандыратындай программа интерфейсін өңдеу;

2) пайдаланушылардың интерфейстері жиі сәтсіздіктерге ұшырайды, себебі олар техникалық немесе функционалдық спецификациялар негізінде пайдаланушылардың өздерінің қатысуынсыз жасалады;

3) пайдаланушылар жайындағы ақпараттарды толық меңгермей «бірінші қолдан» программаны жасау қолданушылардың санының аздығына әкелері анық.

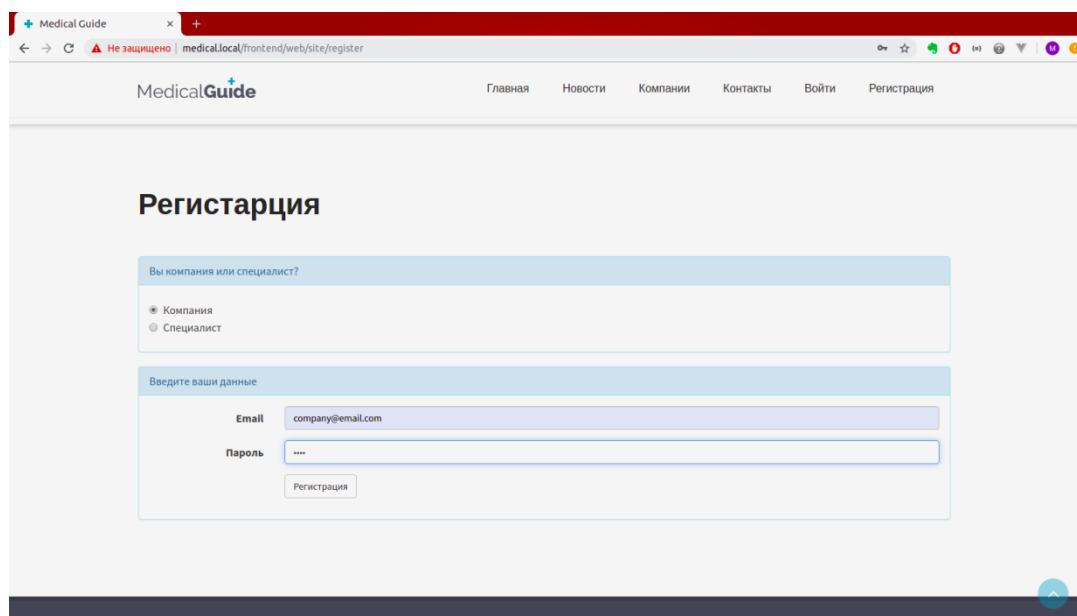
Жоғарыдағы берілген ескертулерді алдыға мақсат ете отырып, жұмыс барысында үнемі пайдаланушылармен қарым-қатынаста болдық. Олардың негізгі талаптары программа максималды түрде қарапайым және түсінікті болуы болды.

Программа негізгі екі блоктан тұрады (мамандарға, яғни жұмысқа қабылдап алушылар, тексерушілер әрі аттестациядан өткізушілер және компанияларға арналған блоктар). Әрбір интерфейснің өзінің мүмкіндіктер жиыны бар. Программаны жіберер алдында пайдаланушылар тіркелуі керек, тіркелгеннен кейін әрбір пайдаланушы өзінің интерфейсiмен сәйкес әрекеттерін орындайды. Негізгі бет 9-суретте келтірілген. Бұл бір жағынан компанияның негізгі терезесі, контактiлері, қосымша байланыстағы компаниялар, өзіңіздің деректеріңіз және т.б. ақпараттарды көруге мүмкіндіктер берітін бет ретінде берілген.



9-сурет – Негізгі бет

Мұнда негізгі бетке кіргеннен кейінгі әрекет ретінде программалық қаматамамен тіркелмей танысу жұмыстарын да жүргізуге болады, ал егер қандай да бір әрекеттерді программалық қамтама арқылы жүзеге асырғымыз келсе, онда міндетті түрде барлық орындалатын операциялар тек тіркелгеннен кейін ғана орындатын болады. Тіркелу бетінің интерфейсі 10-суретте келтірілген.

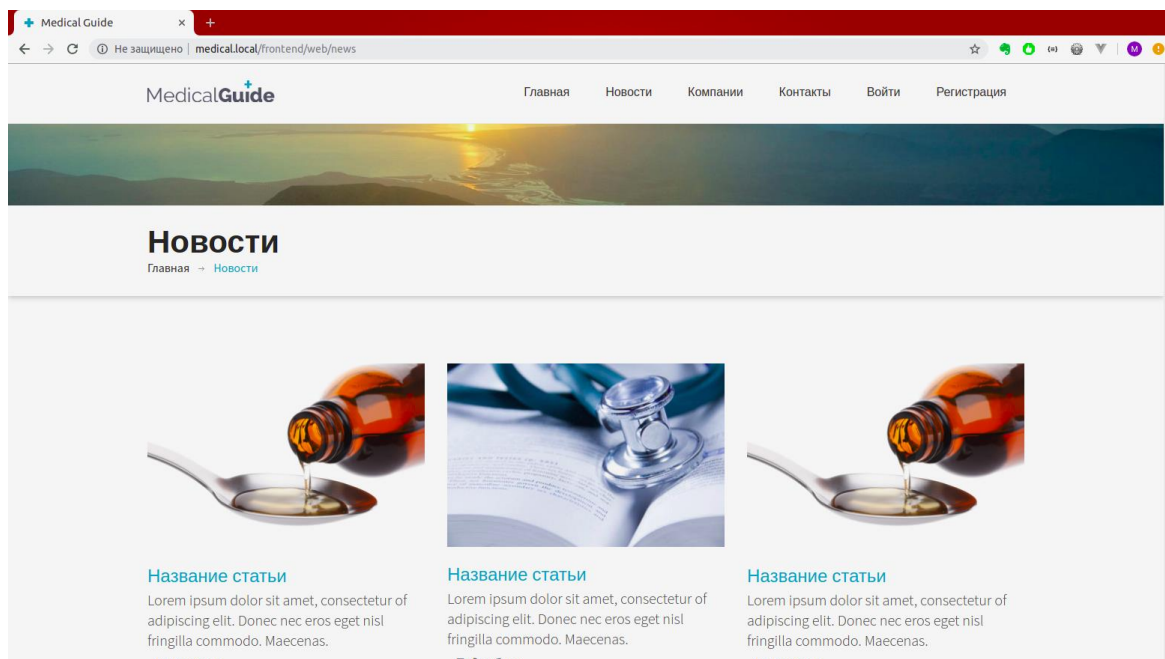


The screenshot shows a web browser window with the URL `medical.local/frontend/web/site/register`. The page header includes the 'MedicalGuide' logo and navigation links: Главная, Новости, Компании, Контакты, Войти, and Регистрация. The main content area is titled 'Регистарция' and contains a registration form. The form has two main sections. The first section, 'Вы компания или специалист?', has two radio buttons: 'Компания' (selected) and 'Специалист'. The second section, 'Введите ваши данные', contains two input fields: 'Email' with the value 'company@email.com' and 'Пароль' with masked characters. Below these fields is a 'Регистрация' button.

10-сурет – Тіркелу беті

3.8.2 Программалау және алгоритмін құру

Алгоритмін құру бизнес-процесстерді толық зерттеп болғаннан кейін жүзеге асырылады және ол бастапқыда құрылған техникалық тапсырмадағы талаптарға сәйкес болуы міндетті болып табылады. Толық бизнес процесстегі орындалатын әрекеттерді түсіну үшін әрі жаңалықтар бетін көру үшін келесі 11-суретпен таныс болыңыздар. Мұнда барлық жаңалықтармен танысып отыруға болады.

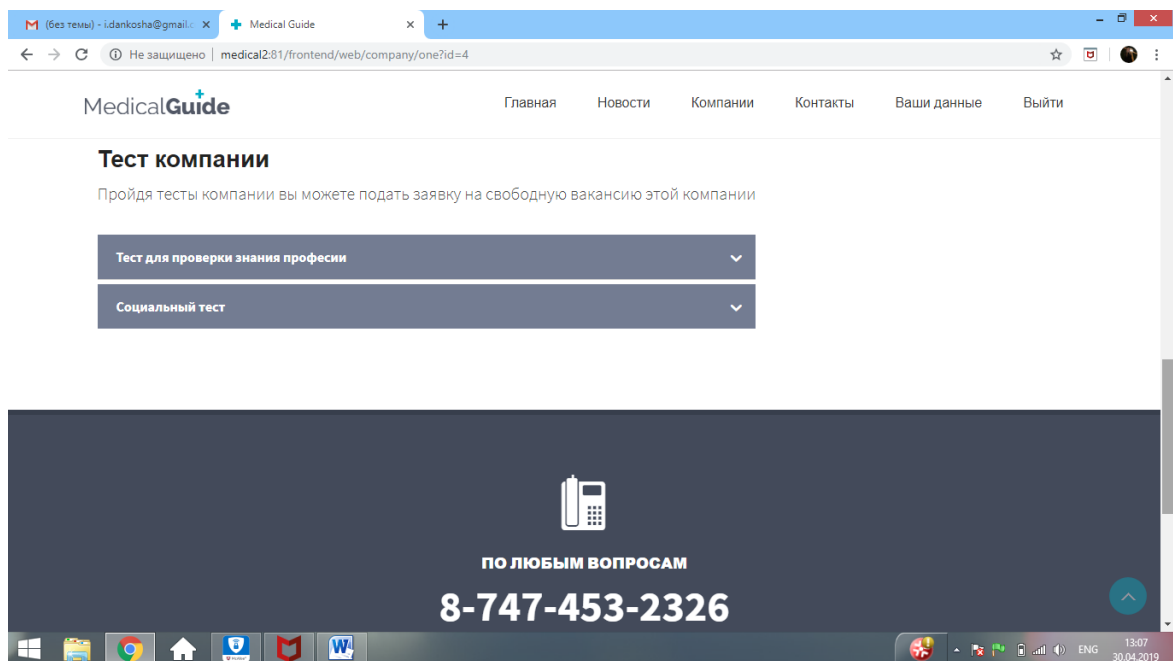


11-сурет – Жаңалықтар беті

3.8.3 Тестілеу

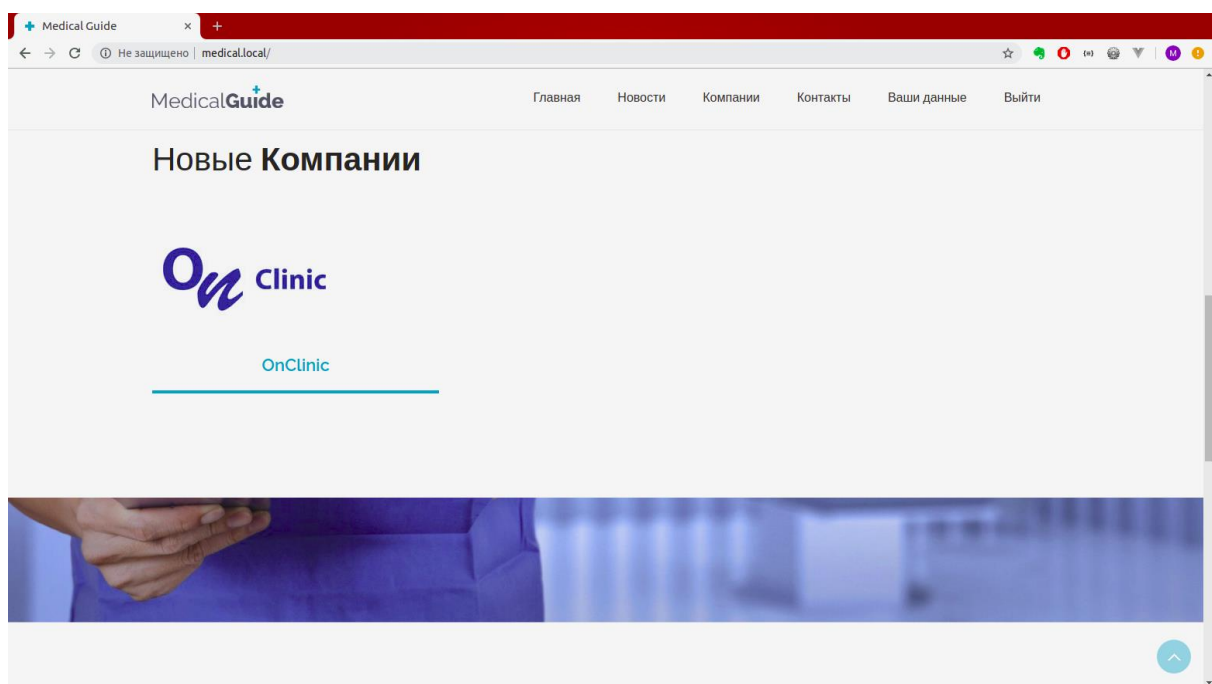
Бұл программалық қамтаманың негізгі әрі маңызды моменттерінің бірі бұл тестілеу, аттестациялау, іріктеу және бағалау әрекеттерін жүргізе алуында болып табылады. Сондықтан бұл әрекеттерді жүргізуі және жүргізілген әрекеттерге сәйкес құжаттарды шығарып бере алуы, немесе сақтап қоюы маңызды орындалатын тапсырмалардың бірі болып табылады.

Персоналды тестілеу үшін тестілеу әдістемесі өңделген, деректер қоры тестілік тапсырмалармен толтырылған және сәйкес құжаттармен қамтамасыздандыра алады. Персоналды тестілеу модулінің интерфейсі келесі 12-суретте келтірілген.

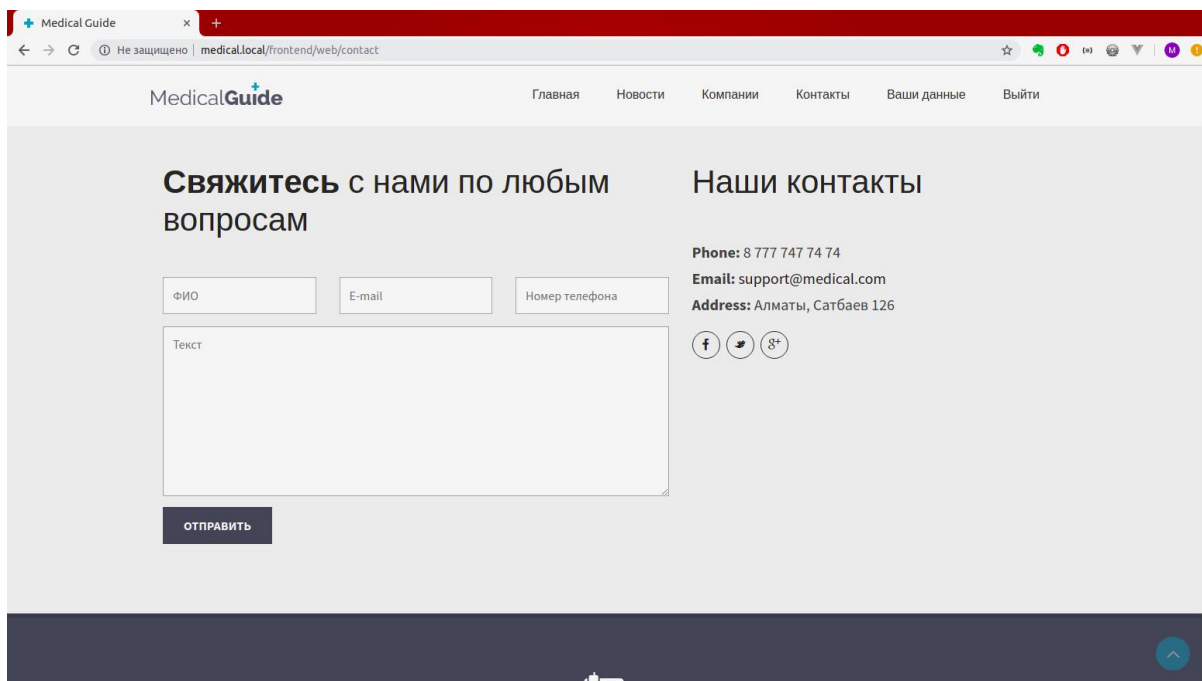


12-сурет – Персоналды аттестациялау, тестілеу және бағалау модулі

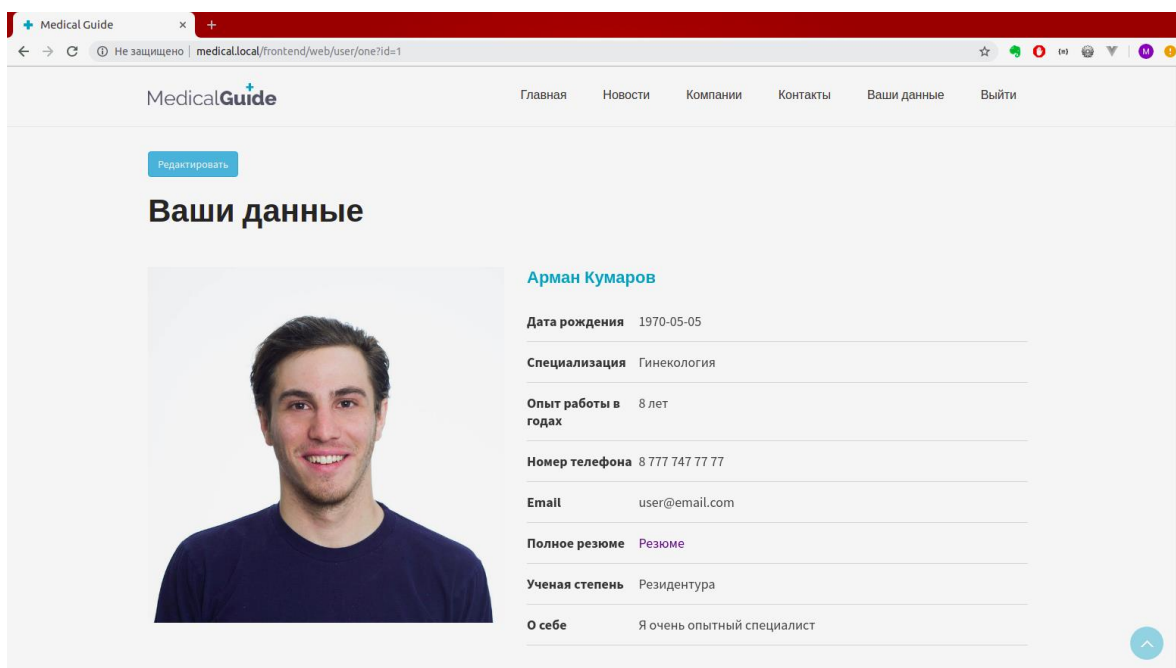
Программалық қамтамаға жаңа компанияларды қосу, байланыс контактілерін беру, өзіңіздің деректеріңізді қарау келесі 12,13,14-суреттерде келтірілген.



12-сурет – Жаңа компанияларды қосу модулі

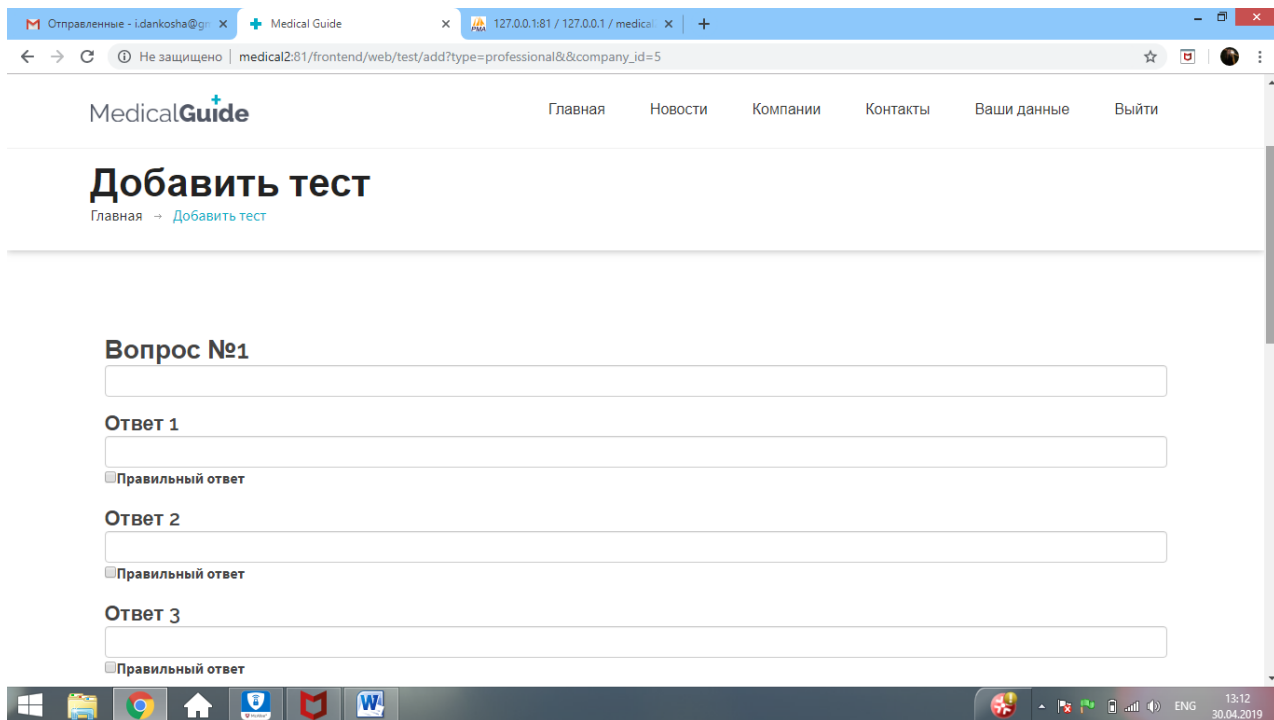


13-сурет – Байланыс контактілерін қарау модулі



14-сурет – Өзі жайындағы ақпараттарды алу модулі

Сонымен қатар әрбір осы сайтқа қосылған компания өзінің жасаған тесітілік сұрақтарын сайтқа енгізу арқылы дектер қорында автоматты түрде сақтау жұмыстарын жасауға мүмкіндіктері болады. Тестілік сұрақтарды сақтау келесі 15-суретте келтірілген.



15-сурет—Тестілік сұрақтарды қосу

ҚОРЫТЫНДЫ

Берілген дипломдық жобаның басты мақсаты персоналды іріктеп таңдап алу жүйесін өңдейтін программалық қамтама жасау болып табылды. Мақсатқа жету кезінде көптеген өнімділігін арттыру үшін басқа да модулдерді өңдеу жұмыстары қарастырылды. Көрсетіліп отырған жүйе аптека жүйелерінің арасында бәсекелестікке сәйкес, лидерлік қасиеттері бар жүйе жасалды деп айтуымызға болады. Бұл жүйе арқылы жоғарыквалификациялы мамандарды іріктеп таңдауды жүргізу жұмысы жеңілдетілді. Сонымен қатар бұл жүйені қызметкерлерді аттестациядан өткізу, бағалау жұмыстарын жүргізу модулдері ретінде де қолданса болады.

Осылайшы, «Аптека» АҚ еңбек өнімділігін арттыруға, әрі барлық тексеру және қабылдау жұмыстарын автоматтандырып жылдам, уақытты үнемдеп жүргізуге қол жеткізуге болады.

Берілген жүйені талдау жұмыстары кемшіліктерін тауып, оны жөндеп, пайдаланушылардың ескертулерін ойларын ескере отырып жасалды. Әрі жұмыс барысында алдыға қойылған барлық мәселелер шешілді:

- персоналды теориялық таңдауды қарастырылды;
- осы мәселе бойынша отандық және шетелдік компаниялардың тәжірибесін талдаулары жасалды;
- өндірістік кәсіпорынның сипаттамасын берілді;
- кәсіпорында кадрларды іріктеп таңдау жүйесін талданды;
- кәсіпорында кадрларды іріктеп таңдау жүйесін жетілдіру жолдары ұсынылды, әрі тестілеуді тек персоналды бағалау үшін емес, сонымен қатар аттестациялау және квалификацияларын арттыру жұмыстары үшін де қолдану жақтары қарастырылды.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

- 1 Армстронг М., Практика управления человеческими ресурсами: учебник. –Спб.:Питер, – 2014 г. – 824 с.
- 2 Базарова Т.Ю., Еремена Б.Л., Управление персоналом: учебное пособие. 2-е изд., переаб.и доп. –М.:ЮНИТИ, – 2002 г. – 560 с.
- 3 Бакиева К.Р., Стратегический анализ аптечного рынка и направления повышения конкурентоспособности аптек //Экономика и менеджмент инновационных технологий. – 2014 г. №6.
- 4 Валиуллина Н.Р., Библиотека: найм персонала: научно-практическое пособие. –М.:Либерей-Бибинформ, – 2010 г. – 128 с.
- 5 Веснин В.Р., Управление персоналом. Теория и практика: учебник. –М.: ТК Велби, – 2011 г. – 688 с.
- 6 Владимирова В.В., Правовые аспекты использования кадровых инструментов в процессе отбора персонала//Кадры предприятия. – 2006 г. –№5.
- 7 Alligator Descartes, Tim Bunce. Programming the PERL DBI; Database Programming with PERL., O'Reilly & Associates, Incorporated. – 2000 г. – 213 с.
- 8 Ролов А.В., Фролов Г.В. Базы данных в Интернете: практическое руководство по созданию Web-приложений с базами данных –М.: Издательско-торговый дом «Русская Редакция». – 2000 г. – 315 с.
- 9 Лаура Томсон, Люк Веллинг. Разработка Web-приложений на PHP и MySQL, Москва «DiaSoft». – 2008 г. – 365 с.
- 10 Кузнецов Максим, Симдянов Игорь, Голышев Сергей. PHP 5. Практика создания Web-сайтов. – Спб.: «БХВ-Петербург». – 2005 г. – 311 с.
- 11 Peter Gulutzan, Trudy Pelzer. SQL-99 Complete, Really, CMP Books. – 1999 г. – 228 с.
- 12 Роберт Шелдон, Джоффри Мойе. MySQL: базовый курс. – М.: «Диалектика». – 2007 г. – 300 с.
- 13 А. Леоненков. Самоучитель UML (2-е издание), БХВ-Петербург. – 2004 г. – 215 с.
- 14 Скотт Хокинс. Администрирование Web-сервера Apache и руководство по электронной коммерции. – М.: «Вильямс». – 2001 г. –367 с.

А қосымшасы (міндетті)

Техникалық тапсырма

А.1.1 Студенттің үлгерілеу жүйесін дамытудың техникалық тапсырмасы

Бұл техникалық жоба жұмысқа кадрларды іріктеп таңдау жүйесінің тиімділігін арттыру бойынша ұсыныстар мен тест тапсырмаларын әзірлеу болып табылады. Бұл жүйені фармацевтика саласының бітіруші студенттері, жаңадан жұмыс қарастырып жүрген қызметкерлер, дәріханада фармацевтика саласында жұмыс жасағысы келетін мамандар пайдаланады деп есептеледі.

Есепке алудың автоматтандырылған жүйесі жас мамандарға және жұмыс іздеп жүрген мамандардың жұмысын жақсартады және жақсы мамандарды қажет ететін компаниялардың жұмысын жеңілдетеді.

А.1.2 Тапсырма

Жүйе мамандардың жұмысқа қабылдану барысында енгізілген мәліметтері бойынша ақпараттарды сақтауға және тест сұрақтарынан өтуде сұрақ және жауаптарын өңдеуге арналған.

А.1.3 Функционалдық талаптар

- Жүйе келесі функцияларды орындау мүмкіндігін қамтамасыз етуі керек:
- жүйеге тіркелу (мамандардың тіркелуі, қабілетті мамандар іздеп жүрген компаниялардың тіркелуі);
 - компаниялар туралы жаңалықтар шығарып отыру;
 - компаниялар мамандар үшін дайындаған тесттерін енгізу;
 - әр маман өзін қызықтыратын компания туралы мағлұмат алып, енгізген тест сұрақтарынан өту;
 - мамандар тесттен өтіп болып нәтежиелерін қарау;
 - тіркелген әр маман иесі өзі туралы мағлұматтарды өзгерте алуы;

А қосымшасының жалғасы

А.1.4 Сенімділікке қойылатын талаптар

Енгізілетін ақпаратты бақылауды қамтамасыз етіңіз. Жүйемен жұмыс істеу кезінде пайдаланушының қате әрекеттерін блоктауды қамтамасыз ету. Сақталған ақпараттың тұтастығын қамтамасыз етіңіз.

А.1.5 Техникалық құралдардың құрамы мен параметрлеріне қойылатын талаптар

Жүйе IBM-үйлесімді дербес компьютерлерде жұмыс істеуі керек. Минималды конфигурация: процессордың түрі - Pentium және одан жоғары;

А.1.6 Ақпараттық және бағдарламамен үйлесімділік талаптары

Жүйе Win32 API интерфейсіне іске асыратын Windows отбасының операциялық жүйелерінде жұмыс істеуі керек.

А.1.7 Бағдарламалық құжаттамаға қойылатын талаптар

Бағдарламаланған бағдарламалық модульдер өздігінен құжатталуы керек, яғни. Бағдарламалардың мәтіндері барлық қажетті түсініктемелерді қамтуы керек.

Бағдарламалық қамтамасыз ету жүйесінде қолданысқа қатысты анықтамалық ақпарат және пайдаланушыға ескертулер болуы керек.

Б қосымшасы
(міндетті)

Бағдарламаның мәтіні

/*негізгі бет*/

```
<div class="sub-banner">
  
  <div class="detail">
    <div class="container">
      <div class="row">
        <div class="col-md-12">
          <div class="paging">
            <h2>Редактировать</h2>
            <ul>
              <li><a href="/">Главная</a></li>
              <li><a href="/frontend/web/company">Компании</a></li>
              <li><a>Редактировать</a></li>
            </ul>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="make-appointment-two">
  <div class="container">
    <div class="row">
      <div class="col-md-12">
        <div class="appointment-form">
          <div class="main-title">
            <h2> Данные</h2>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="panel panel-info ">
    <div class="panel-heading">
      Введите ваши данные
    </div>
    <div class="panel-body">
```

Б қосымшасының жалғасы

```
<form action="/frontend/web/company/save?id=<?=$company-
>id?" method="post" enctype="multipart/form-data">
  <div class="jobs">
    <div class="form-group">
      <label for="title">Название компании</label>
      <input type="text" class="form-control" id="title"
name="Company[title]" placeholder="Название компании" value="<?=$company-
>title?">
    </div>
    <div class="form-group">
      <label for="phone">Телефон</label>
      <input type="text" class="form-control" id="phone"
name="Company[phone]" placeholder="Телефон" value="<?=$company-
>phone?">
    </div>
    <div class="form-group">
      <label for="address">Адрес</label>
      <input type="text" class="form-control" id="address"
name="Company[address]" placeholder="Адрес" value="<?=$company-
>address?">
    </div>
    <div class="form-group">
      <label for="text">О компании</label>
      <textarea class="form-control" name="Company[text]"
id="text" rows="5" value="<?=$company->text?"><?=$company-
>text?"</textarea>
    </div>
    <div class="form-group">
      <label for="image">Логотип</label><br>
      title?" style="width:204px;height:auto;"><br><br>
      <input type="file" class="custom-file-input"
name="Company[file]" id="file" >
      <input type="hidden" name="<?= Yii::$app->request-
>csrfParam; ?>" value="<?= Yii::$app->request->csrfToken; ?>" />
    </div>
    <div class="form-group">
      <label for="exampleInputPassword1">Добавить вакансию</label>
      <button class="add-jobs btn btn-warning"
type="button">Добавить</button>
    </div>
  </div>
</form>
<?php
```

Б қосымшасының жалғасы

```
foreach ($jobs as $index => $job) { ?>
    <div class="form-group" id="job<?=( $index +
10000)?>">
        <input type="text" class="form-control" id="job"
name="jobs[<?=( $index+10000)?>]" placeholder="Название вакансии"
value="<?=$job->title?">
        <br>
        <button class="btn btn-danger delete_button"
type="button" data-id="<?=( $index+10000)?>">Удалить</button>
    </div>
    <?php } ?>
</div>
<button type="submit" class="btn btn-
info">Сохранить</button>
</form>
</div>
</div>
</div>

</div>
</div>
</div>

<script>
$(document).ready(function(){
    var number = 1;
    $(".add-jobs").click(function(e){
        e.preventDefault();

        $(".jobs").append('<div class="form-group" id="job'+number+'">\n' +
            '<input type="text" class="form-control" id="job"
placeholder="Название вакансии" name="jobs['+number+']">\n' +
            '\n<br>' +
            '<button class="btn btn-danger delete_button" type="button" data-
id="'+number+'">Удалить</button>'+
            '</div>');
        number++;
    });
    $('body').on('click', '.delete_button', function (e) {

        e.preventDefault();
        var id = $(this).data("id");
```

Б қосымшасының жалғасы

```
$('#job'+id).remove();

    })

});
</script>

/*негізгі бет жалғасы*/

<!--Start Banner-->

<div class="sub-banner">

    
    <div class="detail">
        <div class="container">
            <div class="row">
                <div class="col-md-12">

                    <div class="paging">
                        <h2><?=$company->title?></h2>
                        <ul>
                            <li><a href="/">Главная</a></li>
                            <li><a href="/frontend/web/company">Компании</a></li>
                            <li><a><?=$company->title?></a></li>
                        </ul>
                    </div>

                </div>
            </div>
        </div>
    </div>

</div>

<!--End Banner-->
<style>
    .member-detail {
        padding : 0;
    }
</style>
```

Б қосымшасының жалғасы

```
<!--Start Content-->
<div class="content">

    <!--Start Team Detail-->
    <div class="member-detail">
        <div class="container">
            <?php
                if (Yii::$app->session->has('user_type')) {
                    if (Yii::$app->session->get('user_id') == $company->id &&
                        Yii::$app->session->get('user_type') == 'company') {
                        ?>
                            <br><br>
                            <a href="/frontend/web/company/edit?id=<?=$company->id?"><button class="btn btn-info">Редактировать</button></a>
                                style="color: white;"
                                class="btn
                            <br><br>
                        <?php
                    }
                }
            ?>
            <div id="demo">
                <div class="row">

                    <div class="span12">
                        <div id="team-detail" class="owl-carousel">

                            <div class="post item">

                                <div class="col-md-12">
                                    <div class="team-detail">

                                        <ul>
                                            <li><span class="title">Номер телефона</span>
                                                <span><?=$company->phone?</span></li>
                                            <li><span class="title">Email</span>
                                                <span><?=$company->email?</span></li>
                                            <li><span class="title">Адрес</span>
                                                <span><?=$company->address?</span></li>
                                        </ul>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

Б қосымшасының жалғасы

```
        </div>
    </div>

    </div>
</div>
</div>
<!--End Team Detail-->
<div class="procedures">
    <div class="container">
        <div class="row">
            <div class="col-md-12">
                <div class="main-title">
                    <h2><?=$company->title?></h2>
                </div>
                <div class="procedure-text">
                    <div class="detail">
                        
                        <?=$company->text?>
                        <div class="title-main">
                            <h4>Свободные вакансии</h4>
                        </div>
                    </div>
                    <ul>
                        <?php
                            $jobs = \common\models\Job::find()-
>where('company_id'=$company->id)->all();
                            foreach ($jobs as $job) { ?>
                                <li><span><?=$job->title?></span></li>
                            <?php }
                        ?>
                    </ul>
                    <div class="col-md-12">
                        <div class="detail top-space">
                            <div class="title-main">
                                <h4>Тест компании</h4>
                                <?php
                                    if (Yii::$app->session->has('user_type')) {
                                        if (Yii::$app->session->get('user_id') == $company-
>id && Yii::$app->session->get('user_type') == 'company') {
                                            ?>
```

Б қосымшасының жалғасы

```
<br><br>
<h5>Профессиональный тест
<?php
$prof_test = \common\models\Test::find()-
>where([
    'company_id' => $company->id,
    'type' => 'professional'
])->one();

    if ($prof_test != null) {
        echo '<a style="color: white;"
href="/frontend/web/test/edit?id='.$prof_test->id.'"><button class="btn btn-
warning">Заменить старый [уже имеется]</button></a>';
    } else {
        echo '<a style="color: white;"
href="/frontend/web/test/add?type=professional&&company_id='.$company-
>id.'"><button class="btn btn-info">Добавить</button></a>';
    }
?>
</h5>
<hr>
<h5>Социальный тест
<?php
$prof_test = \common\models\Test::find()-
>where([
    'company_id' => $company->id,
    'type' => 'social'
])->one();

    if ($prof_test != null) {
        echo '<a style="color: white;"
href="/frontend/web/test/edit?id='.$prof_test->id.'"><button class="btn btn-
warning">Заменить старый [уже имеется]</button></a>';
    } else {
        echo '<a style="color: white;"
href="/frontend/web/test/add?type=social&&company_id='.$company-
>id.'"><button class="btn btn-info">Добавить</button></a>';
    }

?>
</h5>
<?php
```

Б қосымшасының жалғасы

```
    } else {
        ?>
        <p>Пройдя тесты компании вы можете подать
заявку на свободную вакансию этой компании</p>
        <div class="procedures-links">

            <ul id="procedures-faq" class="accordion">
                <li>
                    <div class="link">Тест для проверки
знания профессии<i class="icon-chevron-down"></i></div>

                    <ul
                        class="submenu"
                        style="display:none;">

                        <?php
                            $test = \common\models\Test::find()-
>where("'type='professional' AND company_id=".$company->id)->one();
                            if ($test != null) {
                                $userTest
                                    =
                                \common\models\UserTest::find()->where([
                                    'company_id' => $company->id,
                                    'user_id' => Yii::$app->session-
>get('user_id'),

                                    'type' => 'professional'
                                ])->one();
                                if ($userTest == null) { ?>
                                    <li><span><a
href="/frontend/web/test/run?user_id=<?=$app->session-
>get('user_id')?>&&id=<?=$test->id?>"><button class="btn btn-success">Пройти
тест</button></a></span></li>

                                    <?php } else { ?>
                                        <li><span><button
class="btn">Тест пройден (Ваша оценка: <?=$userTest-
>rating?>%)</button></span></li>

                                        <?php }
                                        ?>

                                    <?php }
                                    ?>
                                </ul>
                            </li>
                        <li>
```


Б қосымшасының жалғасы

```
<div class="link">Социальный тест<i
class="icon-chevron-down"></i></div>

<ul class="submenu" style="display:none
;">
    <?php
    $test = \common\models\Test::find()-
>where("type='social' AND company_id=".$company->id->one());
    if ($test != null) {
        $userTest =
\common\models\UserTest::find()->where([
        'company_id' => $company->id,
        'user_id' => Yii::$app->session-
>get('user_id'),
        'type' => 'social'
    ])->one();
    if ($userTest == null) { ?>
        <li><span><a
href="/frontend/web/test/run?user_id=<?=Yii::$app->session-
>get('user_id')?>&&id=<?=$test->id?>"><button class="btn btn-success">Пройти
тест</button></a></span></li>
        <?php } else { ?>
        <li><span><button
class="btn">Тест пройден (Ваша оценка: <?=$userTest-
>rating?>%)</button></span></li>
        <?php }
        ?>
    <?php }
    ?>
</ul>
</li>
</ul>
</div>
<?php
}
} else {
    echo 'Войдите что бы пройти тесты';
}
?>
```

